MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

A.I. Memo No. 1326 April 1992

Contour Matching Using Local Affine Transformations

Ivan A. Bachelder and **Shimon Ullman**

Abstract

Visual processing tasks often require the matching of contours in two images. Examples include determining image motion and matching features for object recognition. We propose a scheme that takes partial constraints on the matching between contours in two images and finds the matches between these contours using local affine transformations. This new scheme is motivated in part by existing ideas for both recovering optical flow and matching features for object recognition, and in part by short-term motion processing in the primate visual system. The scheme assumes that contours are locally approximated by orthographic projections of planar objects. A local affine transformation is determined for each contour point using oriented elliptical Gaussian neighborhoods that smoothly integrate information over proximally connected contours at several spatial scales. At the largest scale satisfying available constraints a minimal solution mechanism employs a modified pseudoinverse to directly predict matches that are closest to the simplest purely translational correspondence. Using this scheme, the matches for points along contours can be determined in parallel without iteration. The scheme's matching performance is assessed by simulation on noisy synthetic and natural contour imagery.

We thank Eric Grimson and Amnon Shashua for looking over drafts of this paper. This report describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the laboratory's artificial intelligence research is provided in part by National Science Foundation, contract number IRI 8900267, and in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract N00014-85-K-0124.

1 Introduction

At the heart of many visual processing tasks lies the determination of correspondence between features in two images. For example, correspondence is at least implicitly required in stereopsis, structure from motion, and object recognition. Often the matching is underconstrained because the features can be matched in more than one way. Furthermore, the search space for possible matches may be prohibitively large. Assumptions, usually derived from regularities of the physical world, are therefore necessary to constrain matching solutions uniquely. For example, in stereopsis one can constrain the search to small areas along an epipolar line by exploiting uniqueness, continuity, and probabilistic assumptions about disparity [24, 26, 33].

We address the problem of matching contours in a pair of images, assuming partial constraints on the matching are available. As illustrated by Figure 1, we are given: (1) an image containing a set of contours; (2) a second image obtained by displacing, stretching, rotating, scaling, and distorting these contours; and, (3) matching *constraint lines* for several points along the first set of contours constraining the match for each of these points to a line in the second image. The contour images may, for instance, be obtained by applying an edge finding algorithm to either sampled time-varying imagery or two different views of a scene. The constraint lines accompanying these two images can often be estimated for many matching applications. One such application is the measurement of visual motion. In determining optical flow along isobrightness contours, the tangential component of motion frequently must be constrained by additional assumptions due to the *aperture problem* [25, 10, 17, 1, 14]. Local measurements of motion in regions without trackable features (such as edge corners) can capture the component of image velocity only in the direction normal to isobrightness contours. Therefore, the match for a point in one image frame is often constrained at best to a line in the next image frame.

Another problem requiring the matching of contours arises in alignment-based methods for object recognition (for an overview of alignment approaches, see [22, 9, 40, 18, 5]). In these methods, global transformations between model and object views are compensated for by a normalization stage, which aligns the two views and allows subsequent comparison. These views can be represented by contours. Alignment schemes require that object and model features be matched at some stage. Assuming that model and object views can be roughly aligned using global image properties or several pre-matched "anchor points" [18], constraint lines can often be estimated for each object contour point by the tangent at the closest model contour point [4]. (The main inaccuracies in this method will occur for constraint lines at high curvature points, which are simply ignored in the matching process.) A similar method can be used in finding correspondence for apparent motion.

This paper describes a new scheme for solving these contour matching problems.

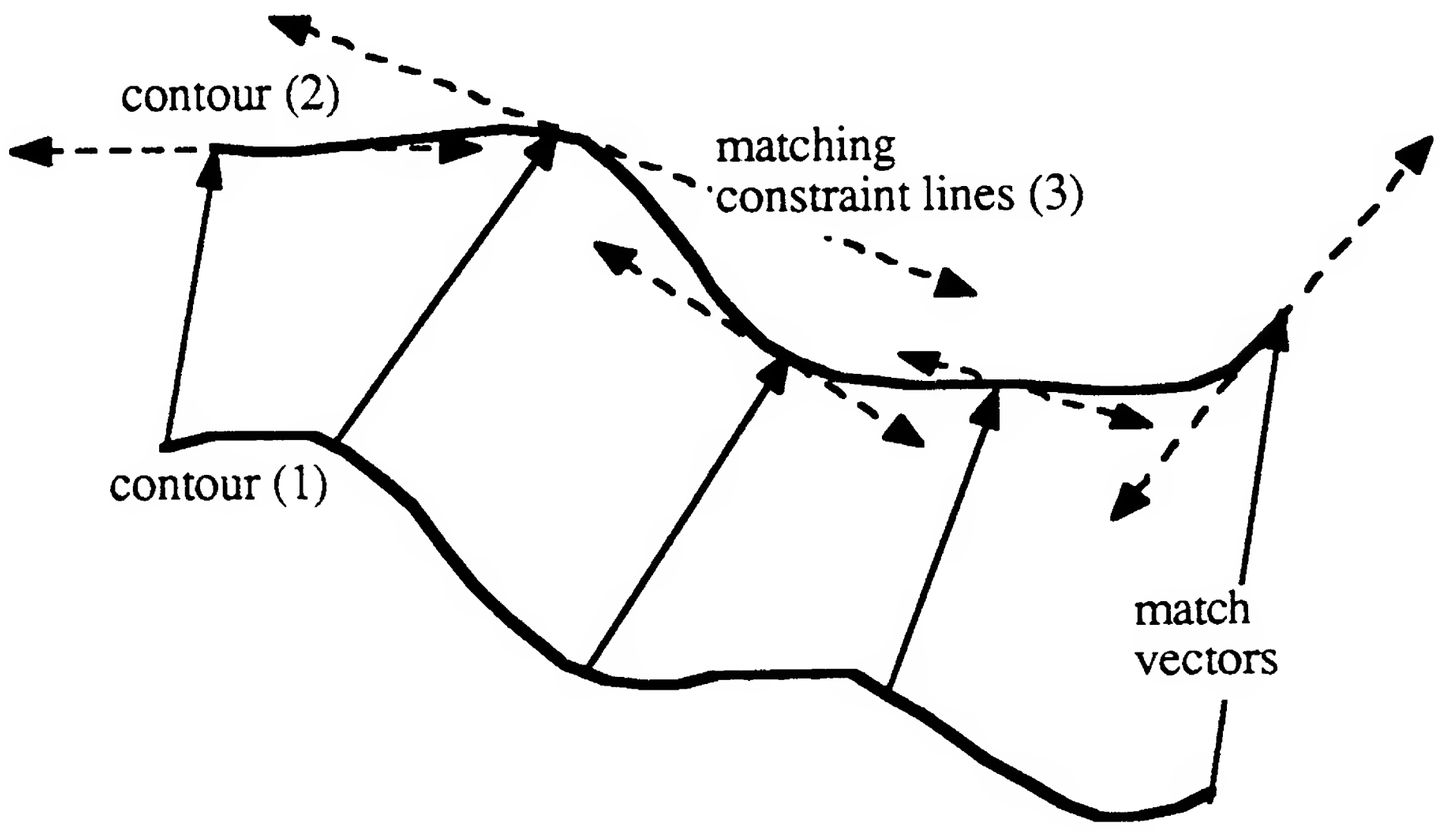


Figure 1: *To find a point-to-point correspondence between contours in two images we are given* **constraint lines** *for several points in the first image. Each constraint line narrows down the match for a point in the first image to a line in the second.*

First, we outline several previous methods for matching contours in the contexts of both recovering optical flow and finding matches for object recognition. Second, we describe the new scheme which uses local affine transformations to predict matches between contours. Next, we briefly review some motivating data concerning the computation of pattern motion in the primate visual system. Finally, we present simulation results obtained on noisy synthetic and natural imagery.

2 Previous Matching Ideas

In computing optical flow, several methods have assumed that the motion of patterns can be described at least locally by pure translation [21, 32, 1]. This assumption can be used to track objects in the case of simple observer motion, but a more general assumption is necessary to account for objects undergoing combinations of translation with three-dimensional rotation, scaling, and deformation. A common approach to the general motion problem is to find the velocity field with the least variation that is consistent with local motion measurements [17, 27, 2, 11, 19]. This idea is based on the observation that objects are locally rigid. Such methods have been applied either in a region-based manner [10, 16], or along image contours [12, 13, 28]. These schemes usually require slowly converging iterative methods, such as the conjugate gradient method, to efficiently recover a global solution constrained by hundreds of linear equations [13]. They also suffer from stability problems at or near the degenerate case of translating linear contours.

A number of methods have assumed that objects in motion can be described by planar patches, an idea originally developed for the recovery of structure from motion [43]. This idea is the basis for the "velocity functional method" of Waxman and Wohn [44]. Their method determines the second order terms of the Taylor series expansion of optical flow by satisfying the normal flow constraints within fixed image neighborhoods. In each neighborhood a residual indicating "goodness of fit" is used to segment the motion field into analytical regions. Because the size of these neighborhoods remains fixed, however, highly non-planar segment boundaries cannot be accurately described. The method also has difficulties with matching contours that are perspective projections of less than biquartic complexity (such as ellipses, parabolas, etc). This singular situation, termed "the aperture problem in the large", primarily arises when the normal flow in the fixed neighborhoods does not adequately constrain the series coefficients. As with smoothness, the solution becomes unstable (sensitive to noise) as objects approach these singular configurations.

Other methods that exploit local planarity include affine motion methods that select local neighborhoods using iterative successive refinement [7, 6]. At a given spatial scale a low pass filter with appropriate bandwidth is applied to the residual motion field. Then, assuming the motion field varies smoothly, the residual motion

field obtained at large spatial scales influences the calculation of the motion field at smaller scales. Although convergence is often quite rapid for nearly planar surfaces in motion, many iterations are required for highly non-planar surfaces undergoing large motions. No attempt is made to shape neighborhoods so that normal flow is integrated primarily along connected contours. Finally, singular conditions similar to those affecting the Taylor series approximation methods again leave even the largest spatial scales unstable.

In object recognition, several contour matching schemes employ correlation measures or examine angles and intersections between contours. Such methods either assume perfect alignment, or conduct a complete search through feature space. They cannot deal with un-parameterized distortions, or inaccuracies prevalent in contour images. They also typically lack efficient implementation. Recently, a number of methods have been devised for comparing contours using affine invariant curvature [8], arc length [3], or moments [15]. Such methods parameterize contours according to one of these measures, normalize the contours by shifting their representation in parameter space, and correlate the resulting invariant representations. The main drawback of these methods is that they are global measures, applicable only to complete curves. As a result, these methods cannot deal with noise or occlusion. Furthermore, as the objects producing the contour image becomes less planar, the affine invariance becomes less valid for describing transformations. Another disadvantage is that the calculation of these affine invariant quantities requires a high degree of differentiation along contours, implying both that contour tracing, contour thinning, and contour enhancement be performed in order to ensure a unique path along the contour, and that contour smoothing be performed in order to avoid large errors in the differentiation due to noise. Finally, these recognition techniques are intended as verification steps taken after a matching has already been hypothesized by exponential search or inaccurate heuristics. This is also true of other affine matching techniques [20, 42].

3 The Proposed Solution

In the scheme described below the transformations of contours within local image neighborhoods are assumed to be affine. We implicitly find the affine transformation through a least squares fit to available match constraint lines and pre-matched points. The affine transformation is established between local contour segments, rather than complete curves (as in [3]), based on the displacement field within a small patch (as in [44]). It constrains the matching by assuming that contours are the orthographic projections of locally coplanar points, thereby reducing the recovery of correspondence to a local, linearly constrained, non-iterative calculation.

Local neighborhoods are constructed so that constraint line information is smoothly integrated primarily over proximal connected contours. This is accom-

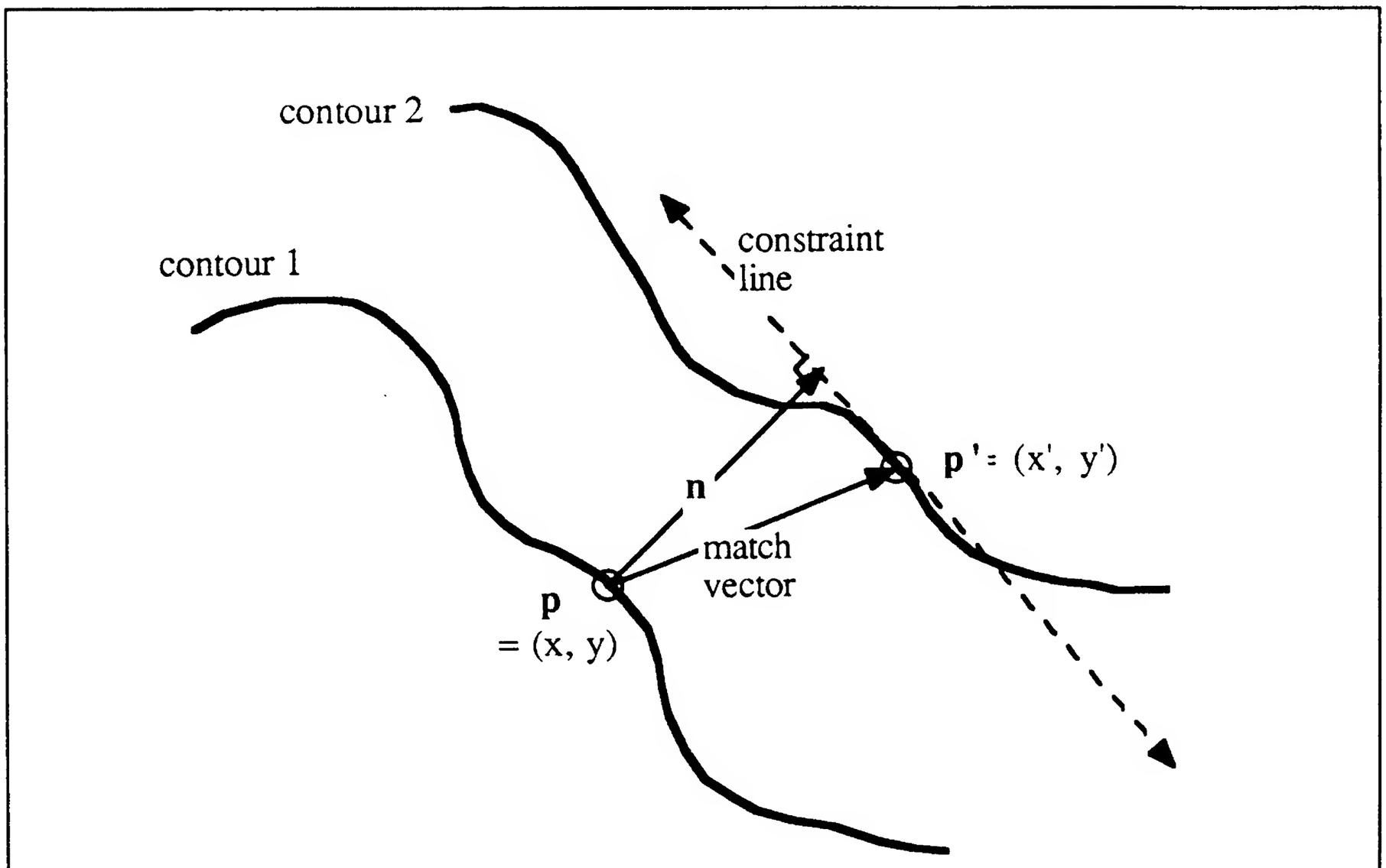


Figure 2: *The unknown match point,* $\mathbf{p}'_i$*, on contour 2 for* $\mathbf{p}_i$ *on contour 1 is assumed to lie along a known constraint line described by a normal vector,* $\mathbf{n}_i$*.*

plished by using elliptical Gaussian neighborhoods oriented along the contour. To choose the appropriate size of the neighborhoods, we consider several neighborhoods of differing sizes simultaneously for a given point. From these neighborhoods, we choose the largest neighborhood within which an affine transformation can accurately satisfy the available constraints. A stable, unique solution is guaranteed for the chosen neighborhood by using a modified pseudoinverse method to find, subject to the constraints, the matches that deviate the least from the minimal purely translational correspondence.

The following sections detail each facet of the scheme in an incremental fashion. We start with the matching of contours that are orthographic projections of rigid planar surfaces.

4 Matching Globally Planar Contours

Let us first assume that the transformation between two image contours can be described by a *global* affine mapping. That is, each contour point $\mathbf{p}_i = (x_i, y_i)$ in the first image maps to contour point $\mathbf{p}'_i = (x'_i, y'_i)$ in the second image by the

linear equation

$$\mathbf{p'}_i = A\mathbf{p}_i + \mathbf{t} \tag{1}$$

where the 2×2 matrix A accounts for the two-dimensional shearing, scaling, and rotation in the image plane about the global origin, $\mathbf{p}_o = (0, 0)$, and the vector $\mathbf{t}$ accounts for two-dimensional translation in the image plane. Next, consider the contours in Figure 2. Suppose that the exact location of $\mathbf{p'}_i$ is unknown, but that it lies along some known *constraint line* in the image. The perpendicular distance between the match point $\mathbf{p}'_i$ and the constraint line should be zero. Therefore, if $\mathbf{n}_i$ is the perpendicular from $\mathbf{p}_i$ to the constraint line, and $\hat{\mathbf{n}}_i$ is the unit normal to the constraint line, then the constraint line is described by

$$(\mathbf{p}_i + \mathbf{n}_i - \mathbf{p'}_i)^T \hat{\mathbf{n}}_i = 0. \tag{2}$$

Substituting (1) into (2) yields

$$(A\mathbf{p}_i)^T \hat{\mathbf{n}}_i + \mathbf{t}^T \hat{\mathbf{n}}_i = \mathbf{p}_i^T \hat{\mathbf{n}}_i + |\mathbf{n}_i|. \tag{3}$$

Since the right hand side of this equation can be computed from available information, we obtain one linear equation constraining the six parameters of the affine transformation for each point. If we represent the six affine parameters as a vector

$$\mathbf{a} = \begin{bmatrix} A_{00} & A_{01} & A_{10} & A_{11} & t_x & t_y \end{bmatrix}^T, \tag{4}$$

then (3) can be rewritten as

$$\mathbf{c}_i\mathbf{a} = d_i \tag{5}$$

where

$$\mathbf{c}_i = \begin{bmatrix} \hat{n}_{i_x} x_i & \hat{n}_{i_x} y_i & \hat{n}_{i_y} x_i & \hat{n}_{i_y} y_i & \hat{n}_{i_x} & \hat{n}_{i_y} \end{bmatrix} \tag{6}$$

$$d_i = \mathbf{p}_i^T \hat{\mathbf{n}}_i + |\mathbf{n}_i|. \tag{7}$$

(Note that $\mathbf{c}_i = \mathbf{0}$ and $d_i = 0$ if either no constraint information is available for point $\mathbf{p}_i$, or $\mathbf{p}_i$ does not lie on a contour.) Thus, for a system of k points and associated equations (where k is the number of points in the image),

$$C\mathbf{a} = \mathbf{d} \tag{8}$$

where C is a $k \times 6$ matrix with rows $\mathbf{c}_1 \ldots \mathbf{c}_k$, and $\mathbf{d}$ is a vector with elements $d_1 \ldots d_k$. When there are more than six independent equations, (8) should be solved in the least squares sense. Specifically, we solve the system

$$C^T C\mathbf{a} = C^T\mathbf{d} \tag{9}$$

which minimizes the mean squared distance between each match and its constraint line. This equation predicts matches for all points, even points without constraint lines, provided that an affine transformation is uniquely determined by existing constraint lines in the image. (Later, we extend this prediction to the underdetermined case.)

5 Incorporating Matched Feature Points

In some matching problems, the exact point-to-point correspondence may be known for some special feature points, such as corners, terminators, high curvature points, inflection points, and isolated points. To incorporate the influence of a feature point match between $\mathbf{p}_i$ and $\mathbf{p'}_i$ we once again assume the match is given by equation (1) and minimize the distance,

$$|(A\mathbf{p}_i + \mathbf{t}) - \mathbf{p'}_i|. \tag{10}$$

Hence, the two constraint equations for each feature point $\mathbf{p}_i$ are given by

$$F_i\mathbf{a} = \mathbf{p'}_i \tag{11}$$

where

$$F_i = \begin{bmatrix} x_i & y_i & 0 & 0 & 1 & 0 \\ 0 & 0 & x_i & y_i & 0 & 1 \end{bmatrix}. \tag{12}$$

(Once again, $F_i = 0$ and $\mathbf{p}'_i = \mathbf{0}$ when $\mathbf{p}_i$ has no given feature point match.) Note that these two equations describe two perpendicular lines that intersect at $\mathbf{p'}_i$. Feature matches can therefore be treated as a special type of contour point that contributes two constraint lines instead of one. Again, for a system of k feature points, we have

$$F\mathbf{a} = \mathbf{g} \tag{13}$$

where F is a $2k \times 6$ matrix consisting of rows $F_1 \ldots F_k$, and $\mathbf{g}$ is the length $2k$ concatenation of $\mathbf{p'}_1 \ldots \mathbf{p'}_k$. Combining (13) with (8) yields

$$\begin{bmatrix} (1-\alpha)C \\ \alpha F \end{bmatrix} \mathbf{a} = \begin{bmatrix} (1-\alpha)\mathbf{d} \\ \alpha\mathbf{g} \end{bmatrix}, \tag{14}$$

where α, a number between 0 and 1, is the accuracy of feature point matches relative to the accuracy of contour point constraint lines. Finally, finding the best affine transformation amounts to solving the least squares relation

$$\left((1-\alpha)C^TC + \alpha F^TF\right)\mathbf{a} = (1-\alpha)C^T\mathbf{d} + \alpha F^T\mathbf{g}, \tag{15}$$

obtained by differentiating (14) with respect to $\mathbf{a}$ and setting the result to zero. Again, solving this system of equations for $\mathbf{a}$ determines the best global affine transformation about the global origin, $\mathbf{p}_o$.

6 Matching Locally Planar Contours

Since contours are generally perspective projections of non-planar, non-rigid objects, (1) cannot in general accurately describe matches globally. The scheme

therefore enforces this affine transformation assumption only locally. That is, the constraint information is integrated within a *local neighborhood* around each contour point, $\mathbf{p}_j$. This local neighborhood is constructed by weighting the constraints at each point.

To fulfill this neighborhood requirement, it is convenient to describe the computations using a local coordinate system at each contour point, $\mathbf{p}_j$. The location of each point, $\mathbf{p}_i$, is measured with respect to the local origin, $\mathbf{p}_j$, instead of the global origin, $\mathbf{p}_o$. To make the calculations local we also weigh constraints for each point, $\mathbf{p}_i$, by some locality measure, ω_i. The "closer" $\mathbf{p}_i$ to $\mathbf{p}_j$, the larger this weight. Employing the method of weighted least squares to incorporate this weighting scheme into (15), we find the local affine transformation, and hence the match, for each contour point, $\mathbf{p}_j$, by satisfying a system of equations

$$R\mathbf{a} = \mathbf{l} \tag{16}$$

where

$$R = (1-\alpha)C^T W^2 C + \alpha F^T W^2 F \tag{17}$$

$$\mathbf{l} = (1-\alpha)C^T W^2 \mathbf{d} + \alpha F^T W^2 \mathbf{g}. \tag{18}$$

The diagonal matrix W establishes the local neighborhood at $\mathbf{p}_j$ and is given by

$$W = \mathrm{diag}(\omega_1 \ldots \omega_k). \tag{19}$$

Note that the resulting 6×6 matrix, R, and the six-dimensional vector, $\mathbf{l}$, can both be written explicitly as summations of point locations, normals, and weights by expanding the matrix definitions. Therefore, the elements of these matrices can easily be calculated in parallel for each local neighborhood using simple adders. In the following sections we consider appropriate ways to shape and scale this neighborhood, thereby defining W.

7 Neighborhood Shape: Oriented Elliptical Neighborhoods

The weight for each point determines the relative influence of its constraints upon the solution. The set of all point weights therefore determines the extent and shape of the local neighborhood. We determine the weights according to several neighborhood criteria. First, the neighborhood integration should monotonically decrease with distance from the local origin, since distant points are less likely to be coplanar with $\mathbf{p}_j$. Second, the neighborhood should be smooth so that matching solutions vary continuously along contours. (Note, however, that this will not mandate the *smoothest* solution along the contour.) Finally, the neighborhood should be maximally elongated and oriented along the contour in order to integrate

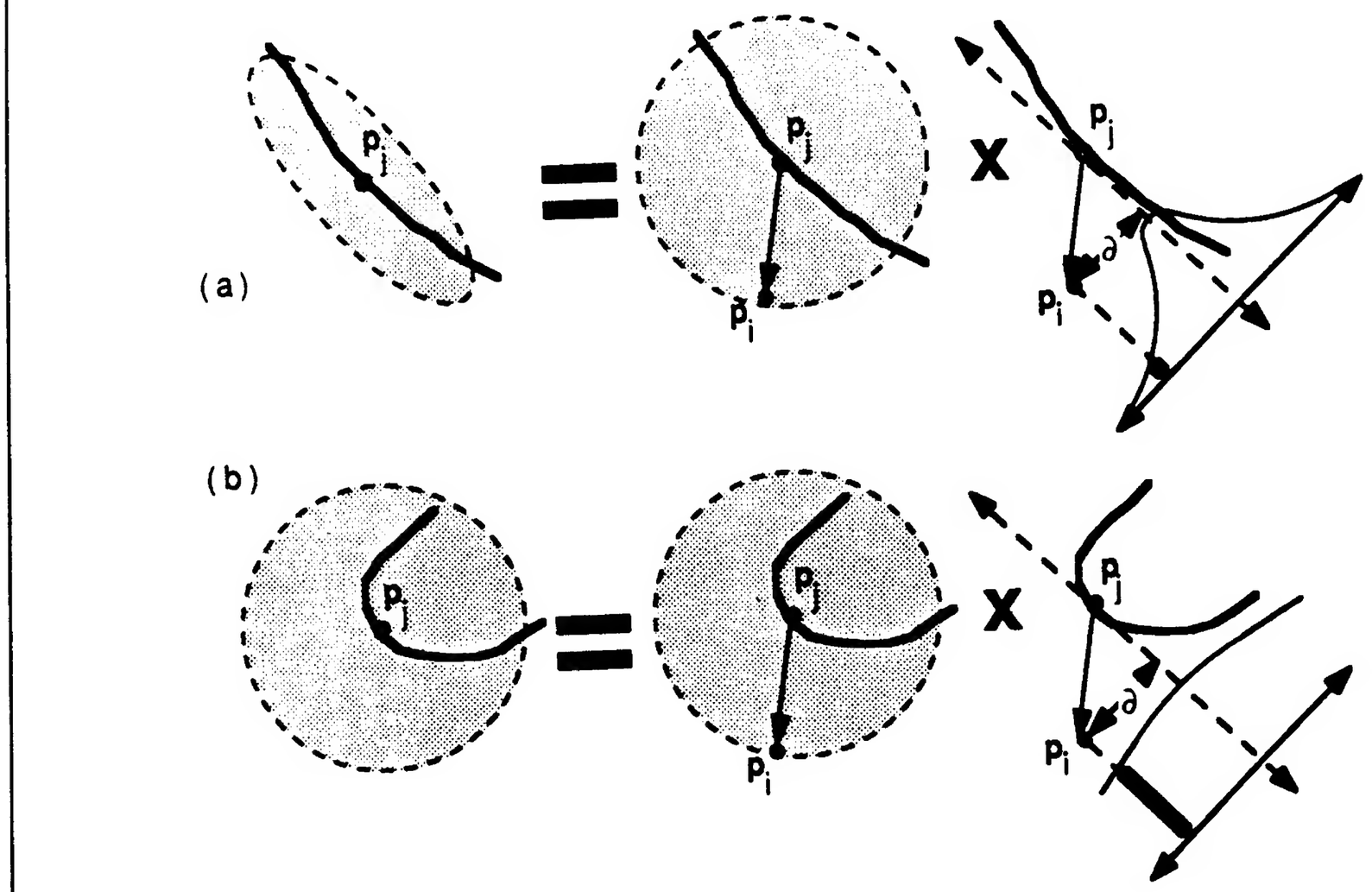


Figure 3: *(a) Integration along nearly linear contour segments is primarily along the contour, while (b) integration over symmetric sections of the contour is circular. Oriented local neighborhoods are determined by the product of γ_i, a Gaussian of the distance from the local origin $\mathbf{p}_j$, and μ_i, a Gaussian of the distance from the axis of local orientation passing through $\mathbf{p}_j$. The width of the latter Gaussian is determined by the strength of the local orientation.*

information primarily along connected proximal contours without serial contour tracing.

With these criteria in mind, we suggest that the weight for $\mathbf{p}_i$ actually be the product of two Gaussian weights:

$$\omega_i = \gamma_i \mu_i. \tag{20}$$

The first weight, γ_i, is given by

$$\gamma_i = \exp\left(-|\mathbf{p}_i|^2 / 2\sigma_{size}^2\right). \tag{21}$$

The set of all γ's constructs a circularly symmetric Gaussian neighborhood about the local origin. The standard deviation of this Gaussian, σ_{size}, is the effective size of this neighborhood. This parameter is discussed in the next section. The second weight, μ_i, is given by

$$\mu_i = \exp\left(-\delta_i^2 / 2\sigma_{shape}^2\right) \tag{22}$$

where δ_i is the perpendicular distance from $\mathbf{p}_i$ to the axis of local orientation passing through $\mathbf{p}_j$. It modulates the circularly symmetric Gaussian and orients the resulting elliptical neighborhood to integrate information primarily over connected contours without explicitly performing connectivity analysis or contour tracing. The width of this Gaussian, σ_{shape}, ranges between 0 and ∞, corresponding to maximal local orientation and circular symmetry respectively. Hence, the stronger the local orientational preference of the contour, the higher the aspect ratio of the elliptical neighborhood, and the larger the relative influence of points that lie closer to the local axis of orientation. This neighborhood construction is illustrated in Figure 3.

We now determine σ_{shape} (assuming first that we know σ_{size}) keeping in mind that the oriented neighborhood must be narrow for linear contours and wide for circularly symmetric contours. To capture this notion, we propose that the major and minor axes of the elliptical neighborhood be respectively aligned with and proportional (by some proportionality constant β) to the major and minor axes of the principal component ellipse of the local binary contour image. The principal component ellipse can be derived through principal component analysis of the inertia matrix

$$J = \begin{bmatrix} J_{xx} & J_{xy} \\ J_{xy} & J_{yy} \end{bmatrix} \tag{23}$$

where

$$J_{xx} = \sum_{i=1}^{k} \gamma_i^2 b_i x_i^2, \qquad J_{xy} = \sum_{i=1}^{k} \gamma_i^2 b_i x_i y_i, \qquad J_{yy} = \sum_{i=1}^{k} \gamma_i^2 b_i y_i^2 \tag{24}$$

are the second local moments about the local y, $-xy$, and x axes respectively, and b_i is the value of the binary contour image at $\mathbf{p}_i$. First, the second moment extremes are the eigenvalues of J,

$$\lambda_{max} = \frac{J_{xx} + J_{yy} + \rho}{2}, \qquad \lambda_{min} = \frac{J_{xx} + J_{yy} - \rho}{2} \tag{25}$$

where

$$\rho = \sqrt{(J_{xx} - J_{yy})^2 + (2J_{xy})^2}. \tag{26}$$

These eigenvalues give the axes magnitudes of the principal component ellipse, to which the neighborhood axes should be proportional. If the major axis of the neighborhood is $\sigma_{size} = \beta\lambda_{max}$, then the minor axis is simply $\beta\lambda_{min} = \frac{\lambda_{min}}{\lambda_{max}}\sigma_{size}$. This desired minor axis is the effective width of the modulated Gaussian in the direction perpendicular to the axis of orientation; that is,

$$\frac{\lambda_{min}}{\lambda_{max}}\sigma_{size} = (\sigma_{size}^{-2} + \sigma_{shape}^{-2})^{-\frac{1}{2}}. \tag{27}$$

Rearranging, we obtain

$$\sigma_{shape} = \frac{\lambda_{min}/\lambda_{max}}{\sqrt{1 - (\frac{\lambda_{min}}{\lambda_{max}})^2}}\sigma_{size}. \tag{28}$$

Next, δ_i is given by the scalar product of $\mathbf{p}_i$ and the unit normal to the axis of local orientation, defined by the unit eigenvector of J corresponding to the eigenvalue λ_{max}:

$$\hat{\mathbf{\Theta}}_\perp = \frac{\left[\sqrt{\rho + \text{sign}(J_{xy})(J_{yy} - J_{xx})}, \quad -\text{sign}(J_{xy})\sqrt{\rho + \text{sign}(J_{xy})(J_{xx} - J_{yy})}\right]}{\sqrt{2\rho}}. \quad (29)$$

Hence,

$$\delta_i = \hat{\mathbf{\Theta}}_\perp^T \mathbf{p}_i \quad (30)$$

(see [16] for a more complete analysis of using the principal component method to find the orientational preference of binary images).

Note that, although there may be problems with finding the orientation of a contour that is surrounded by several other contours, surrounding contours should not in general affect the final solution long as we have chosen a sufficiently small neighborhood in which to carry out the calculations. This issue is addressed next.

8 Neighborhood Size: Choosing From Multiple Neighborhood Scales

For several reasons it is desirable to include in the computation a mechanism for selecting the size of each integration neighborhood. As the size of the local neighborhood decreases, the local planarity assumption is better approximated, and the possible interference from nearby contours decreases. However, if the neighborhood is too small, lack of adequate constraints on the local affine transformation produces an unstable solution. Clearly there is a tradeoff between extracting sufficient information for unique solutions within large neighborhoods, and satisfying local planarity assumptions and insulating computations from interfering contours with small neighborhoods.

To solve this problem we use a number of spatial scales computed in parallel. Out of several possible solutions obtained for different neighborhoods (several values of σ_{size}) around $\mathbf{p}_j$, the solution corresponding to the largest spatial scale that best satisfies (16) prevails. There are several possible ways to determine how well a given spatial scale uniquely satisfies the constraints. For instance, one could use the least squares residual to determine how well the assumptions fit the local neighborhood. Using this method, we solve equation (16) at several scales and choose the largest scale for which the least squares residual is small. Alternatively, one could use the condition number of the matrix R to determine if a given local scale is too small. Under this criterion we take the solution at the *smallest* spatial scale for which the condition number of R is lower than some number, κ_{max}.

Finally, we could choose the spatial scale based on some combination of these two methods.

The condition number criterion is adopted here for the sake of efficiency in serial simulation. First, this criterion allows us to try smaller scales, which require less integration time, before larger ones. Second, it does not require that we actually solve (16) at each scale, as does the least squares residual. In setting the maximum condition number, κ_{max}, one must consider the accuracy of the constraint lines and feature points. If these quantities are known to be accurate, then κ_{max} should be high to promote precise solutions. If, however, the accuracy is low, then κ_{max} should be low to decrease the solution's sensitivity to such errors.

9 Minimal Solutions for Ambiguous Cases

Despite selection of the optimal neighborhood, the solution to (16) can be underdetermined or unstable, even in the largest neighborhood. This situation occurs when there exists more than one affine mapping between two contours. Consider, for example, a linear contour where the constraint lines for all points are identical. In this case any combination of scaling or shearing along the constraint line, in addition to any translation taking the line to the constraint line, satisfies the constraints. Another example is the matching of concentric circles. If at each point along the first circle the constraint line is parallel to the tangent to the circle at that point, then any amount of rotation, coupled with an appropriate scaling, is possible (see examples in Figure 4).

For these singular cases, we select the *minimal solution*. Intuitively, the minimal solution, chosen from the set of solutions which satisfy the available constraints, is closest to the smallest purely translational matching. This idea is based entirely on intuition. First, it is conceivable that many object transformations not described by a unique affine transformation (a first order transformation) can be described by a unique translation (a zero-th order transformation). Second, when even the translational matching is ambiguous, it seems that the solution should be closest to the normal component of the match to reflect the lack of constraints. These notions are illustrated in Figure 4. Any amount of rotation of the circle, or any amount of tangential translation of the line, results in a set of match vectors that deviate more from their normal components.

In formulating the mechanism for choosing the minimal solution defined above, we must retain some important features. First, the mechanism must uniquely (stably) determine matches, regardless of the quantity or quality of the constraint information. Second, the mechanism must preserve continuous matches along contours. Finally, the minimal solution must have a closed form solution that is efficiently calculated at each point without iteration.

One way to satisfy all these requirements is to minimize the cost function

$$\Delta = \sum_{i=1}^{k} (\omega_i |A\mathbf{p}_i + \mathbf{t} - (\mathbf{p}_i + \mathbf{t}_{min})|)^2, \tag{31}$$

which summarizes over the entire neighborhood the squared deviation between predicted match vectors and $\mathbf{t}_{min}$, the smallest pure translation that satisfies neighborhood constraints in the least squares sense. Even for collinear points with parallel constraint lines (which can be matched by many purely translational transformations), Δ is uniquely minimized by the average neighborhood normal component. Furthermore, Δ is robust to noise because it integrates information over the entire local neighborhood, as opposed to directly minimizing the deviation between each match and the normal component, which is sensitive to noise in the constraint line. As for smoothness, calculation over Gaussian local neighborhoods guarantees continuous variation of Δ. Finally, Δ is computed independently of (in parallel with) the final solutions at other points, as opposed to global selection methods which select the set of contour matches to optimize an overall measure, such as smoothness along the contour.

Assuming $\mathbf{t}_{min}$ has been computed (see later section), minimizing Δ is identical to finding the affine transformation that matches feature points (the feature matches for each point $\mathbf{p}_i$ simply being $\mathbf{p}_i + \mathbf{t}_{min}$). Therefore, to minimize Δ we solve a system of linear equations, similar to those of (13), given by

$$Q\mathbf{a} = \mathbf{h} \tag{32}$$

where

$$Q = S^T W^2 S, \qquad \mathbf{h} = S^T W^2 \mathbf{v}. \tag{33}$$

Here S, similar to F, is a $2k \times 6$ dimensional matrix with rows $S_1 \ldots S_k$, where

$$S_i = \begin{bmatrix} x_i & y_i & 0 & 0 & 1 & 0 \\ 0 & 0 & x_i & y_i & 0 & 1 \end{bmatrix}, \tag{34}$$

and $\mathbf{v}$, similar to $\mathbf{g}$, is a length $2k$ concatenation of $\mathbf{v}_1 \ldots \mathbf{v}_k$, where

$$\mathbf{v}_i = \mathbf{p}_i + \mathbf{t}_{min}. \tag{35}$$

To minimize Δ subject to the matching constraints we solve

$$\min_{\mathbf{a}} (\mathbf{a}^T Q\mathbf{a} - 2\mathbf{a}^T \mathbf{h}) \text{ subject to } R\mathbf{a} = \mathbf{l}. \tag{36}$$

Once again, $\mathbf{t}_{min}$ is the smallest least squares neighborhood pure translation, which we have not yet mathematically defined. In the next section we simplify the existing constraints in order to directly compute this default, and eventually the match vector itself, later in the paper.

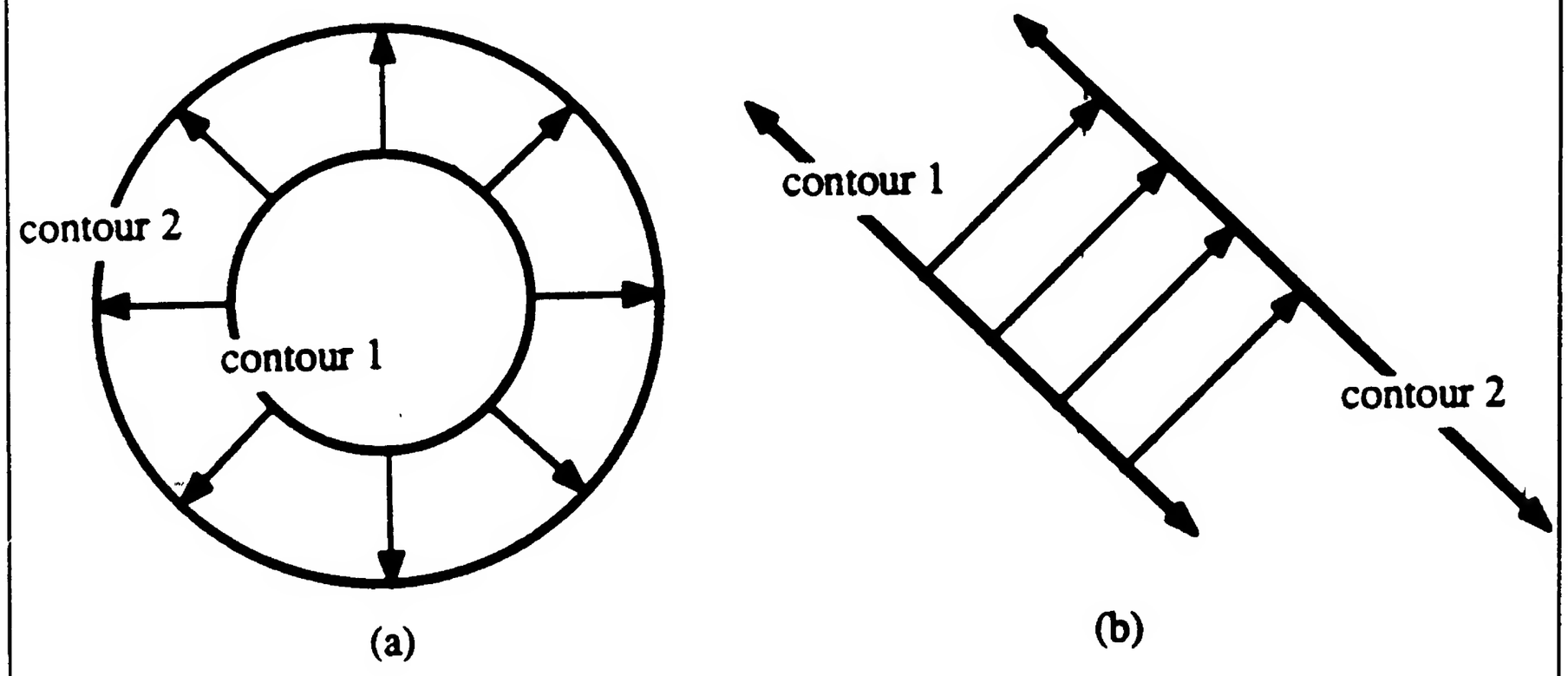


Figure 4: *Minimal solutions for ambiguous transformations are intuitive: (a) concentric circles could be matched with any amount of rotation, but pure expansion should be the default; (b) parallel lines could be matched using a translation in any direction within 90 degrees of the normal component, but the default should be the normal component.*

10 Simplifications Arising from an Implicit Formulation

Before deriving closed form solutions for $\mathbf{t}_{min}$ and the match vector, it is useful to digress and consider how we will eventually obtain final matches for contour points given the computed local affine transformations. This analysis will simplify subsequent computations.

It turns out that a point's match depends only on the translational component of its local affine transformation. That is, the final match for each local origin, $\mathbf{p}_j$, is given by

$$\mathbf{p'}_j = A\mathbf{p}_j + \mathbf{t} = A \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \mathbf{t} = \mathbf{t}. \tag{37}$$

Therefore, we need only implicitly solve for the rotation, scaling, and shearing components and thereby directly compute the match vector. In fact, an implicit formulation should be more efficient and stable. Increased efficiency follows from inverting 2×2 and 4×4 submatrices of the 6×6 R matrix instead of inverting R itself. Increased stability follows from analysis of the elements of R: since the coefficients of $\mathbf{t}$ do not depend on the size of the local neighborhood, while the coefficients of A grow as the square of the local neighborhood, uniform random noise affects these elements differently. By separating the translational components, this undesirable property should disappear.

To separate the components, first let (16) be rewritten as

$$\begin{bmatrix} R_r & R_c \\ R_c^T & R_t \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ \mathbf{t} \end{bmatrix} = \begin{bmatrix} \mathbf{l}_r \\ \mathbf{l}_t \end{bmatrix} \tag{38}$$

where R_t is a 2×2 matrix relating the translational affine coefficients, $\mathbf{t}$, to the 2-dimensional vector, $\mathbf{l}_t$, R_r is a 4×4 matrix relating the rotational, shearing, and scaling affine coefficients, $\mathbf{r}$, to the 4-dimensional vector, $\mathbf{l}_r$, and R_c is a 4×2 coupling matrix. Likewise, Q and $\mathbf{h}$ can be represented by

$$Q = \begin{bmatrix} Q_r & Q_c \\ Q_c^T & Q_t \end{bmatrix}, \qquad \mathbf{h} = \begin{bmatrix} \mathbf{h}_r \\ \mathbf{h}_t \end{bmatrix}. \tag{39}$$

At this point, it is advantageous to expand and simplify several of these submatrices. Expanding, we obtain

$$Q_r = \begin{bmatrix} Q_e & 0 \\ 0 & Q_e \end{bmatrix}, \qquad Q_t = \left(\sum_{i=1}^{k} \omega_i^2 \right) I, \qquad Q_c = 0 \tag{40}$$

$$\mathbf{h}_t = \sum_{i=1}^{k} \omega_i^2 \begin{bmatrix} x_i + \mathbf{t}_{min_x} \\ y_i + \mathbf{t}_{min_y} \end{bmatrix} = \left(\sum_{i=1}^{k} \omega_i^2 \right) \mathbf{t}_{min}. \tag{41}$$

(taking advantage of the fact that $\sum_{i=1}^{k} \omega_i^2 x_i = \sum_{i=1}^{k} \omega_i^2 y_i = 0$ for elliptical neighborhoods due to odd symmetry), where

$$Q_e = \sum_{i=1}^{k} \omega_i^2 \begin{bmatrix} x_i^2 & x_i y_i \\ x_i y_i & y_i^2 \end{bmatrix} \tag{42}$$

and I is the 2×2 identity matrix. Note that the elements of Q_e are second moments taken within the elliptical neighborhood. With this in mind, these second moments should be roughly proportional, by some factor η, to the second moments, J_{xx}, J_{xy}, and J_{yy}, that were used to derive this neighborhood (see (24)). Therefore, a reasonable estimation is

$$Q_e \approx \eta J. \tag{43}$$

All of these simplifications will allow us to solve directly for the default translation. They will also guide the subsequent derivation of the match vector itself.

11 Determining The Default Translation

What remains to be seen, before deriving a closed form expression for the match, is how the default pure translation is determined. Using (38), finding $\mathbf{t}_{min}$ for the neighborhood is much like finding the best local affine transformation with the rotational, shearing, and scaling components set to zero (*i.e.* $A = I$, the 2×2 identity matrix), or

$$\mathbf{r} = \mathbf{r}_{min} = \begin{bmatrix} 1 & 0 & 0 & 1 \end{bmatrix}^T. \tag{44}$$

Thus, we simply solve a system of equations

$$R_t \mathbf{t}_{min} = (\mathbf{l}_t - R_c^T \mathbf{r}_{min}). \tag{45}$$

Since the determination of a purely translational matching merely requires the intersection of two constraint lines, this system of equations is underconstrained only when there are no feature point matches and the constraint lines are parallel. In this case, we choose the solution closest to the average normal component by finding, using the general pseudoinverse (denoted by $+$), the smallest translation satisfying the constraints. Hence,

$$\mathbf{t}_{min} = R_t^+ (\mathbf{l}_t - R_c^T \mathbf{r}_{min}). \tag{46}$$

The pseudoinverse formulation used for this and all subsequent calculations is presented in Appendix B. It differs from the conventionally used form in several ways. First, to promote stability, the SVD threshold is chosen such that the maximum allowed condition number, κ_{max} (introduced previously), mandates the absolute minimum eigenvalue for a given matrix. Second, values below this threshold are not immediately deemed singular, but rather continuously default to zero

as they drop below the threshold. Without this latter modification, matching solutions along contours may vary discontinuously, defeating the smoothing properties of the Gaussian neighborhoods and producing rather non-intuitive results.

12 Direct Match Computation

Finally, we are now in a good position to solve directly for the match vector. Making use of the separation of affine components and the simplifications made possible by this separation, (36) is equivalent to

$$\min_{\mathbf{t},\mathbf{r},\mathbf{\Gamma}_t,\mathbf{\Gamma}_r} \begin{pmatrix} \mathbf{r}^T(Q_r\mathbf{r} - 2\mathbf{h}_r) + \left(\sum_{i=1}^k \omega_i^2\right)\mathbf{t}^T(\mathbf{t} - 2\mathbf{t}_{min}) \\ \quad + \mathbf{\Gamma}_r^T(R_r\mathbf{r} + R_c\mathbf{t} - \mathbf{l}_r) + \mathbf{\Gamma}_t^T(R_t\mathbf{t} + R_c^T\mathbf{r} - \mathbf{l}_t) \end{pmatrix} \tag{47}$$

where $\mathbf{\Gamma}_t$ and $\mathbf{\Gamma}_r$ are the Lagrange multipliers for the translational and rotational parts of the constraints respectively. Taking the partial derivatives with respect to $\mathbf{\Gamma}_t$, $\mathbf{\Gamma}_r$, $\mathbf{t}$ and $\mathbf{r}$ and setting them to zero respectively leaves

$$R_c^T\mathbf{r} + R_t\mathbf{t} = \mathbf{l}_t \tag{48}$$

$$R_r\mathbf{r} + R_c\mathbf{t} = \mathbf{l}_r \tag{49}$$

$$\mathbf{t} = \frac{\mathbf{x}_t}{\sum_{i=1}^k \omega_i^2} + \mathbf{t}_{min} \tag{50}$$

$$\mathbf{r} = Q_r^{-1}\mathbf{x}_r + \mathbf{r}_{min}, \tag{51}$$

where

$$\mathbf{x}_t = -\frac{1}{2}(R_c^T\mathbf{\Gamma}_r + R_t^T\mathbf{\Gamma}_t), \qquad \mathbf{x}_r = -\frac{1}{2}(R_r^T\mathbf{\Gamma}_r + R_c\mathbf{\Gamma}_t), \tag{52}$$

since $Q_r^{-1}\mathbf{h}_r = \mathbf{r}_{min}$, the default for rotational, shearing, and scaling components.

Finally, substitution of (50) and (51) into (48) and (49) then allows us to solve for $\mathbf{x}_t$ and derive

$$\mathbf{t} = (R_t - R_c^T R_r^{\#} R_c)^+ \begin{pmatrix} \mathbf{l}_t - R_c^T R_r^{\#}\mathbf{l}_r \\ \quad - (R_c^T R_r^{\#} R_c - R_t)\mathbf{t}_{min} \\ \quad - (R_c^T R_r^{\#} R_r - R_c^T)\mathbf{r}_{min} \end{pmatrix} + \mathbf{t}_{min} \tag{53}$$

where

$$R_r^{\#} = Q_r^{-1}(R_r Q_r^{-1})^+ = \begin{bmatrix} J^{-1}|J| & 0 \\ 0 & J^{-1}|J| \end{bmatrix} \left(R_r \begin{bmatrix} J^{-1}|J| & 0 \\ 0 & J^{-1}|J| \end{bmatrix} \right)^+ \tag{54}$$

(using the approximation of (43)) and

$$J^{-1}|J| = \begin{bmatrix} J_{yy} & -J_{xy} \\ -J_{xy} & J_{xx} \end{bmatrix}. \tag{55}$$

Equation (53) not only minimizes the overall deviation between neighborhood matches and the best pure translation, but it also guarantees a unique, stable match regardless of the constraints or the neighborhood. When R_r and Q_r are non-singular, then $R_r^{\#} = R_r^{-1}$, and the default, $\mathbf{r}_{min}$, factors out of the solution. Otherwise, one must use the pseudoinverse. The latter case occurs when the neighborhood contour points are collinear, since at least three non-collinear point matches are required to uniquely define an affine transformation. When this situation occurs, Q_r affects the solution by altering the eigenvalues or R_r. It essentially transforms the Euclidean affine parameter space to one in which finding the closest affine solution vector to the purely translational affine transformation,

$$\mathbf{a}_{min} = \begin{bmatrix} \mathbf{r}_{min} \\ \mathbf{t}_{min} \end{bmatrix}, \tag{56}$$

yields an affine transformation that minimizes Δ.

This completes the development of the matching scheme. Applications will undoubtedly need to extend or modify certain details. For instance, the development here, though robust to noise in the constraint lines, says nothing about how such lines are determined. It is important to note, however, that the scheme has been developed in a general context and is therefore applicable any problem requiring that full correspondence between contours in two images be found given only partial matching constraints.

13 Biological Plausibility

The scheme developed in the previous sections is motivated in part by a number of psychophysical and neurophysiological findings regarding the recovery of short-range motion in the primate visual system. Evidence of varying detail indicates that the primate motion system may employ several components of the matching scheme, including the affine assumption, the influence of feature points, local Gaussian neighborhoods, oriented elliptical neighborhoods, multiple spatial scales, and minimal solution mechanisms.

Positive evidence for the affine assumption is sketchy. Interestingly, however, neurons have been discovered in area MST of the macaque monkey that seem to be selectively responsive to divergence [34], deformation [34, 29], and rotation [35] of the visual field, respectively equivalent to the scaling, shearing, and rotational transformations that can be described by an affine transformation. Although the role of these neurons is not entirely clear, they may be involved, along with area MT neurons selective to pure translation [1], in the computation of a local affine approximation to the flow field. Psychophysical experiments indicating that humans can more easily estimate the structure of a moving wire-frame from only two apparent motion frames when the projections undergo an affine transformation

can also be taken as weak evidence for the visual system's employment of this approximation [39].

Terminator feature points have been shown to increase the perception of rigid motion when added to a moving linear contour with a single inflection point [31], much like specially matched feature points affect the determination of an affine transformation here. Furthermore, the perception of rigid contour movement increases as the distance between a terminator and the inflection point decreases, suggesting that the visual system employs some smooth local integration process, such as the Gaussian neighborhoods proposed here. This observation for the most part rules out any type of globally calculated smoothness assumption [12, 13], since the smoothest velocity field for a purely translating contour is a pure translation regardless of terminator locations.

Further results obtained using the same contour stimulus have also indicated that moving terminators placed off the contour can weakly influence the perception of motion, an effect that can be blocked by non-moving terminators placed between the moving terminators and the inflection point [31]. This result supports the notion that motion measurements are integrated primarily along contours by oriented receptive fields, much like the oriented neighborhoods employed here. Not surprisingly, oriented receptive fields have been measured in area MT with estimated dimensions 15 arc min $\times$ 5 arc min in central vision (see [29]).

Parallel short-range motion analysis by several spatial scales is also supported by psychophysical evidence. Several results indicate that both input to and integration within the motion system consists of several spatial scales [1, 41, 29]. Moreover, experiments involving the motion of random dots have indicated that perception of global flow, as opposed to local independent motion, increases as the global distribution of dots becomes more coherent [45] (see [36]). This result suggests that the visual system applies an appropriate assumption to the largest possible area of the image, much like the mechanism for choosing the largest neighborhood that satisfies matching (motion) constraints.

Finally, evidence for minimal solution mechanisms in the visual system can be gathered from experiments regarding the perception of non-rigid motion, which for the most part occurs when the perception of pattern motion is similar to the normal velocity measurements rather that the actual motion field. Perception of coherent pattern motion of two moving sine wave gratings has been found to decrease with the angle between their primary directions [1]. Similarly, the perceived rigidity of a translating sine wave contour (of the form $a \sin(\alpha x + \beta t)$) decreases as the wave becomes more shallow (the angle at the intersection of the curves with the x axis is less than 15 degrees) [30]. Given these results, it has been suggested that the visual system pays more attention to the normal velocity in such highly unstable (ambiguous) situations in order to boost the signal to noise ratio [31]. This suggestion is consistent with the minimal solution mechanism proposed here,

which effectively minimizes the summed squared deviations between neighborhood matches and the average neighborhood normal component when the best purely translational matches are ambiguous.

14 Implementation and Results

The matching scheme was implemented in C on a Sun workstation. Five spatial scales were employed. The sizes, in terms of the variance of the local Gaussian neighborhood, σ_{size}, were 4, 8, 16, 32, and 64 pixels. These sizes were chosen so that the largest neighborhood was on the order of the size of the examples, thereby ensuring that the largest local neighborhood could roughly include the constraints of the entire example. Input consisted of two 256 $\times$ 256 pixel binary contour images. Some were synthetically produced, while others were extracted zero crossings of smoothed and differentiated natural imagery [23]. Though the method should be able to deal with any set of contour images with partially constrained matches, the two images chosen for each simulation were relatively aligned to reflect the fact that constraint lines are most easily derived from mildly differing imagery (*i.e.* time sampled imagery for short-range motion, or roughly aligned objects for object recognition). The "correct" matches between each pair of images were known for synthetic imagery and hand-picked for natural imagery. The constraint line for each contour point was determined by finding the local orientation of the contour, then taking the line with the same orientation that passed through the "correct" match. In the context of visual motion, these constraint lines roughly mimicked a local measurement of normal velocity and allowed comparisons to be made with the results of previous methods for recovering optical flow.

To assess the affect of noise upon the matching process, 10% random noise was added to the components of the normals describing the constraint lines and feature point matches. Assuming this equal noise distribution, a value of 0.5 was used for α, the accuracy of feature point matches relative to the accuracy of constraint lines. Furthermore, the parameter κ_{max}, used both for selecting the size of the neighborhoods and for setting the SVD threshold in pseudoinverse operations, was experimentally determined by applying the scheme to the worst case (most ambiguous) matching problem, the matching of parallel lines (see example 1b in appendix A). The average percentage deviation of matches with respect to the actual normal components of the match (the expected minimal solution) was determined for several values of κ_{max}. Using the results, shown in Figure 5, the tradeoff between stability (low κ_{max}) and accuracy (high κ_{max}) was balanced by choosing $\kappa_{max} = 74.0$, the highest value for which the average percentage error did not exceed the percentage noise. This value was used for all subsequent examples, the results of which are shown in Appendix A. Note that this choice seems particularly appropriate considering the large increase in matching error as κ_{max}

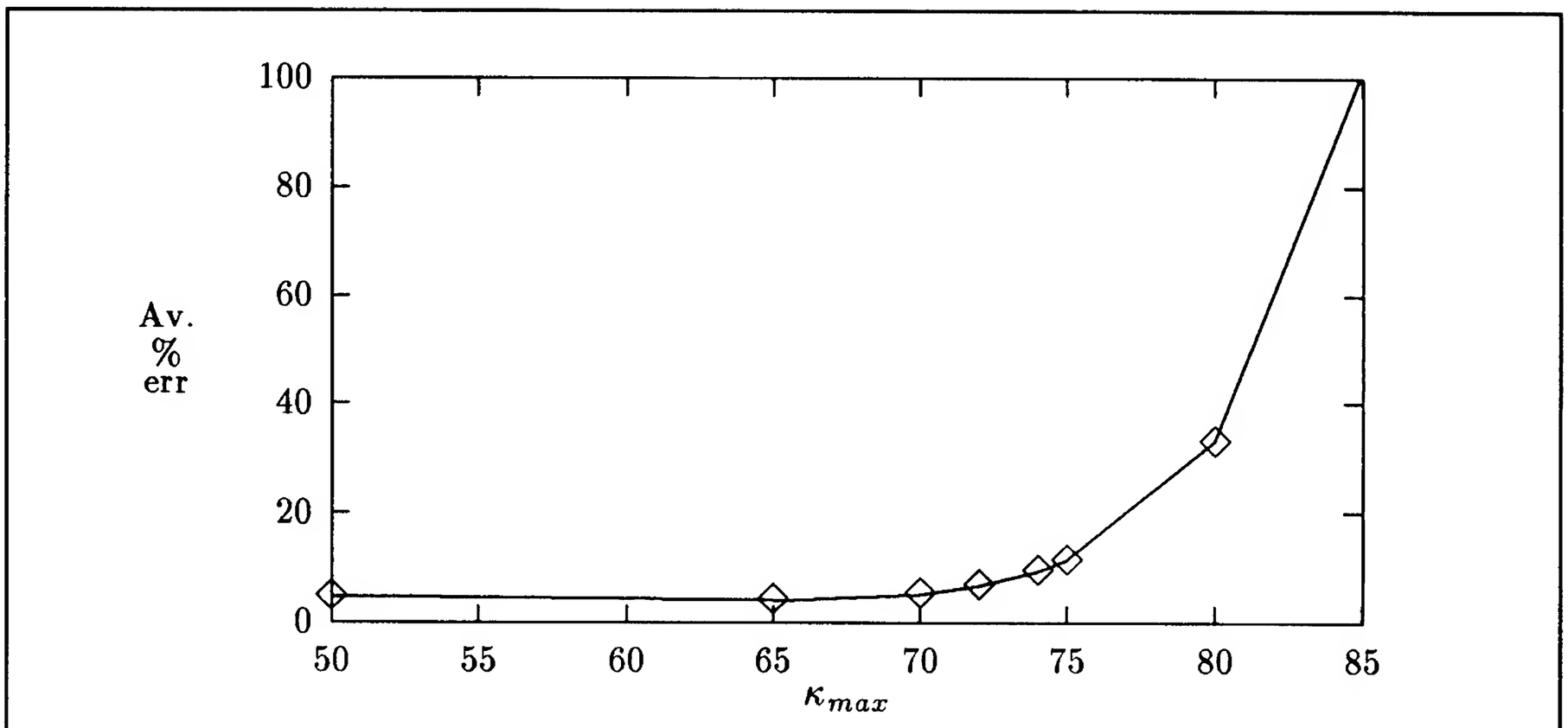


Figure 5: *Average percent matching error vs.* κ_{max} *for the matching of parallel lines with* 10% *random noise in the constraint lines.*

increases beyond this value.

15 Discussion

As demonstrated by the results presented in Appendix A, the proposed contour matching scheme robustly recovers the correspondences of both synthetic and natural contour imagery. First, unlike most other methods for matching contours, this scheme yields intuitive solutions for ambiguous cases, such as parallel lines and rotated ellipses. Moreover, matching solutions continuously approach stable defaults as the matching becomes more ambiguous. For example, as a translated sine wave becomes more linear (smaller amplitude), the solution defaults gradually to the normal components of the match. In the context of short-range motion, this behavior agrees with psychophysical results indicating the non-rigid perception of small amplitude translating sine waves [30] (see earlier section on biological plausibility). A similar default occurs as a rotated ellipse becomes more circular, where the solution again agrees with psychophysical data indicating that the smaller the aspect ratio of the rotating ellipse, the less accurate the final velocity field and the more non-rigid the interpretation. Note that the smoothness assumption [12, 13] yields similar results, except near the two highest curvature points of the ellipse, where the matching is more accurate. More attention must be paid to human perception to determine which of these interpretations is more psychophysically accurate.

A second attribute of the scheme concerns the usefulness of specially matched

feature points. While feature point matches obviously aid the matching process of the ambiguously matched parallel lines, the scheme did not exploit any feature point matches for subsequent examples and nevertheless recovered the actual correspondences. This observation indicates that, while feature point matches are especially important for the matching of ambiguous examples, they are no more powerful than redundant constraint lines when used in conjunction with an unambiguous example.

A third observation is that analysis at several neighborhood scales greatly enhances the scheme's ability to recover exact matches, particularly those collective matches not modeled precisely by a global affine transformation. For example, the three-dimensional distortions used to generate the space curve did not present a problem to the matching scheme. The attribution of this capability to several neighborhood sizes is not obvious from the results, but instead from simulations that reported the sizes of the selected neighborhoods at every contour point. Matching the views of the face, for example, required small neighborhoods for matching contours with intricate detail (such as the eyes and nose) and large neighborhoods for matching nearly linear contours (such as the occluding boundary between the face and the background). It is not until we examine the matches predicted for the three-dimensional rotated wire frame that we see the limitations of multiple scales. Though the overall matching solution is quite accurate, the matching of overlapping contours undergoing separate motion is highly inaccurate because information has in this case been integrated over independent contours.

16 Conclusion and Future Work

A method for determining matches between contours in two images has been proposed, developed, and tested, assuming that constraint lines, each narrowing down the match for a contour point in the first image to a line in the second image, are available for several contour points in the first image. Suggested applications include the determination of optical flow in short-range motion and the matching of aligned contour views in either alignment-based object recognition or long-range motion.

To determine the match for a contour point we find the best affine transformation, in a weighted least squares sense, that satisfies the match constraint lines and feature point matches within an oriented elliptical neighborhood. This neighborhood is established by weighing the constraint equations. The weight for a given point constraint is the modulation of the Gaussian distance to the local origin, which establishes a circularly symmetric local neighborhood, by the Gaussian distance from the axis of local orientation, which attempts to limit the neighborhood to a single contour. The width of the modulating Gaussian is set such that the axes of the elliptical neighborhood are proportional to the local axes of inertia.

To determine the width of the circularly symmetric Gaussian, the effective size of the local neighborhood, we consider several sizes simultaneously and choose the smallest one which yields a stable, unique solution, as determined by the condition number of the constraint matrix, R. In the event that this condition number exceeds the parameter κ_{max} even at the largest neighborhood, there is more than one possible solution, from which we choose the smallest affine transformation which predicts the set of neighborhood matches which deviate the least from the smallest least squares purely translational matching. Unique determination of both the smallest pure translation and the final match is guaranteed through the use of a modified pseudoinverse operation.

This entire procedure is repeated for every contour point. Note, however, that the match for each point may be computed in parallel. Furthermore, since the predicted match for a point is simply given by the translational component of the local affine transformation, finding the match involves explicitly solving for only the two translational components of the affine transformation. A closed form solution for the match involves using a continuous version of the pseudoinverse to invert a 2 $\times$ 2 matrix and a 4 $\times$ 4 matrix, the coefficients of which are weighted summations of local point constraint parameters that may be determined in parallel.

The matching scheme only relies on two parameters. The parameter, α, which ranges between 0 and 1, indicates the expected accuracy of feature point matches relative to constraint line information. The parameter κ_{max}, which must be set according to how much noise is expected in the constraint lines, places a bound on the condition number of any constraint matrix.

As for biological plausibility, more experimentation is necessary to compare the performance of the scheme with that of the visual system, in both motion perception and object recognition. However, the scheme is motivated by current biological evidence for short-range pattern motion and can, in many cases, explain such evidence better than other models can.

Simulation results show that the scheme performs well for most examples, including noisy synthetic imagery and edges extracted from natural imagery. Minimal solutions obtained when the recovery is ill-posed are intuitive, and agree with psychophysical data. It seems that the non-rigidity of contour transformations arises primarily near ambiguous situations, and incorporation of terminators improves the correspondence significantly for such near-ambiguous examples.

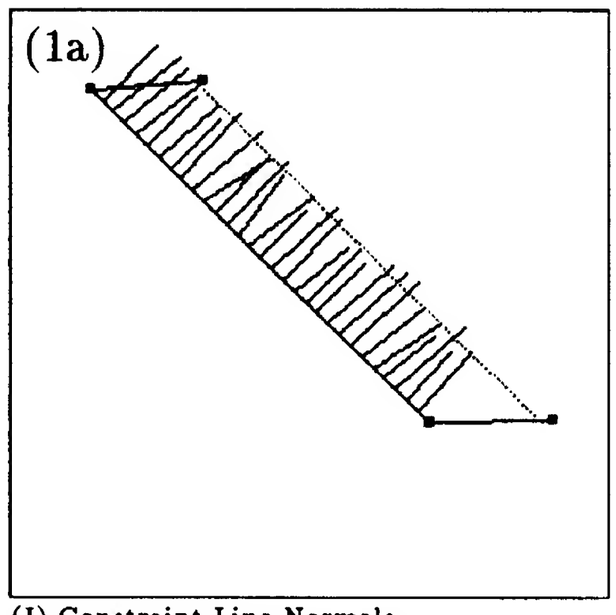
Despite the scheme's performance and biological plausibility, further research remains. First, the precise conditions under which the local affine transformation is uniquely determined by the constraints should be better understood. Currently, it is difficult to predict when the minimal solution mechanism is necessary to guarantee uniqueness without examining specific cases individually. Second, simulations should more rigorously test matching capabilities and limitations. Specifically, experiments should test the theoretical effectiveness of using oriented elliptical

neighborhoods to limit the interaction of independent contours. They should also assess matching performance upon examples larger than the largest neighborhood. With this data, the optimal number and sizes of available neighborhoods can be suggested. Third, application specific problems can be addressed. In recovering optical flow, it should be straightforward to test the scheme using extracted normal velocities from actual motion frames. In alignment-based recognition, pre- and post-processing steps for finding the constraint lines and final matches should be developed and tested. Finally, extensions and modifications to the method can be considered. An extended scheme which makes use of available depth information to form ellipsoidal neighborhoods, where the three-dimensional Gaussian distances are used to weight point constraints, may effectively deal with the current inaccurate matching caused by overlapping, independently moving contours. Finally, prediction mechanisms such as Kalman filtering techniques can be incorporated for applications, such as recovering optical flow, that are aided by temporal improvement of continuous matching solutions over multiple image frames.

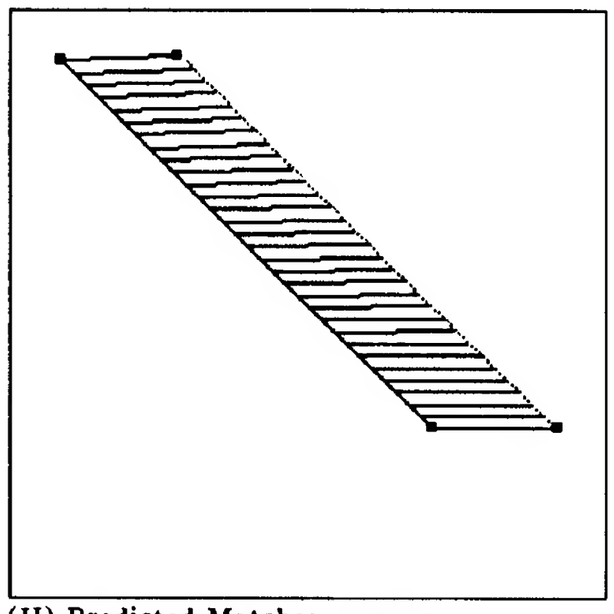
A Simulation Results

In each example presented the first contour image is shown in black, and the second contour image is superimposed as dotted contours. In column *I*, vectors describing the constraint lines are shown for sampled points along the first contour, and special match vectors are indicated by small squares at the endpoints. 10% random noise has been added to these vectors. The computed and actual match vectors for sampled points are then shown in columns *II* and *III*, allowing qualitative comparison. Quantitative results are also presented. At each sampled point, the relative error is computed by normalizing the distance between predicted and actual matches by the length of the actual match vector. The average relative error, computed over all sampled points, is reported for each example. Note that predicted matches do not necessarily lie exactly upon the second contour, especially for the matching of ambiguous examples. In practice, final matches may be found by taking the point on the second contour that is closest to the predicted match.

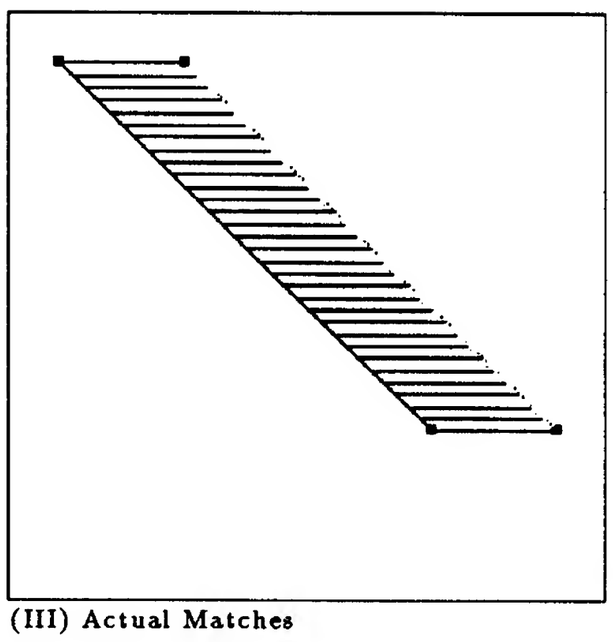
Rows 1a and 1b respectively present the matching of synthetic parallel lines with and without terminator matches. Rows 2a and 2b respectively present the matching of translated synthetic large and small amplitude sine waves. Next, row 4a presents the matching of a synthetic ellipse rotated in the image plane by 20 degrees, and row 4b presents the matching of synthetic rotated concentric circles. In row 4a an orthographic projection of a synthetic wire-frame is matched for purely translational correspondence, and in row 4b the same wire-frame is rotated by 20 degrees about each axis and matched. Row 5 presents the matching of an orthographic projection of an arbitrary synthetic 3D space curve rotated by 10



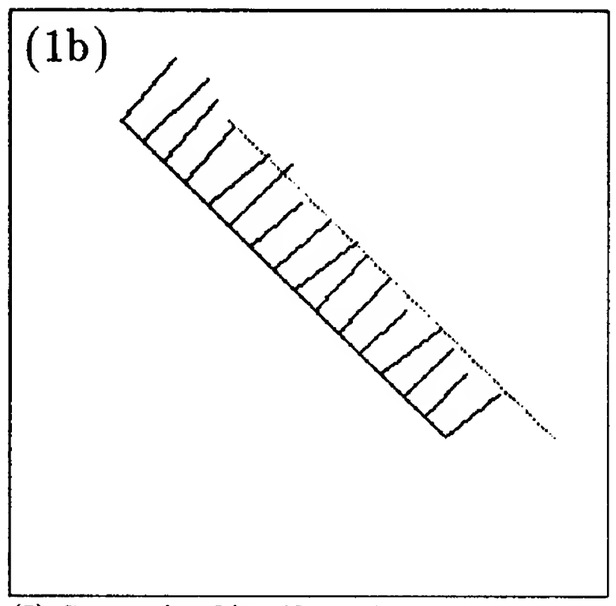
(I) Constraint Line Normals
(with 10% noise)



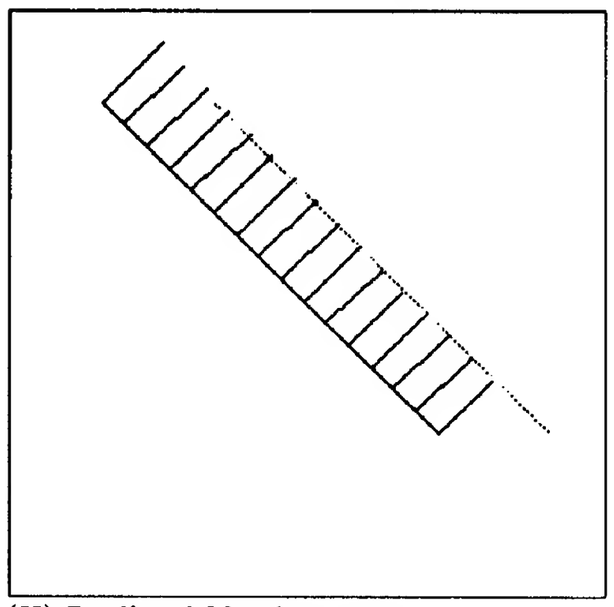
(II) Predicted Matches
(average percent error: 3.26%)



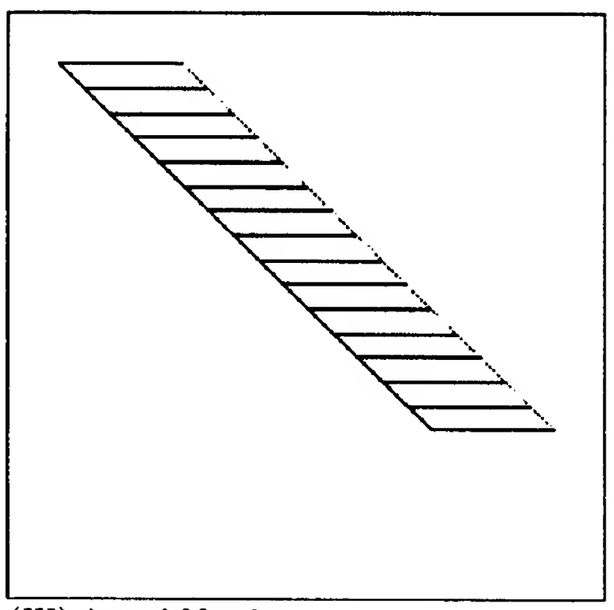
(III) Actual Matches



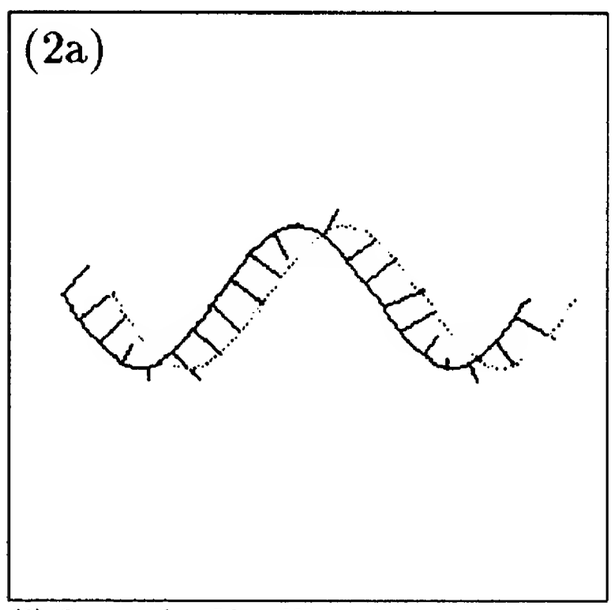
(I) Constraint Line Normals
(with 10% noise)



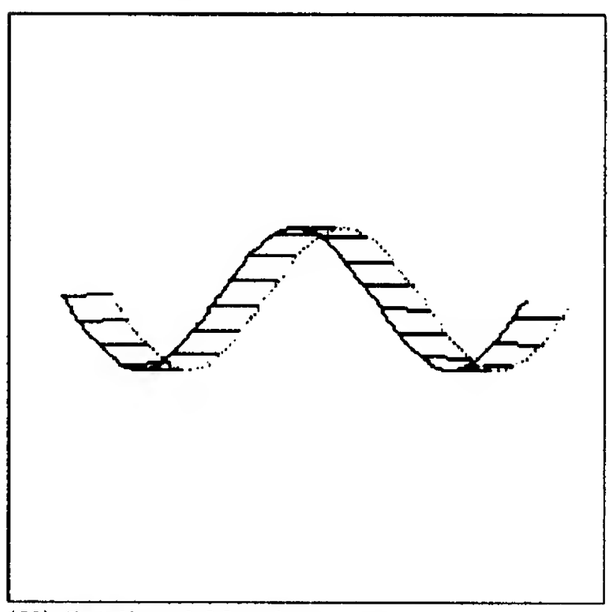
(II) Predicted Matches
(average percent error: 69.6%)



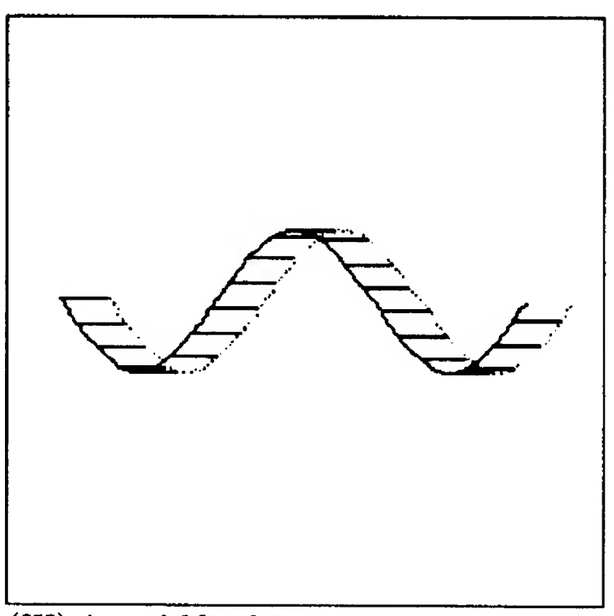
(III) Actual Matches



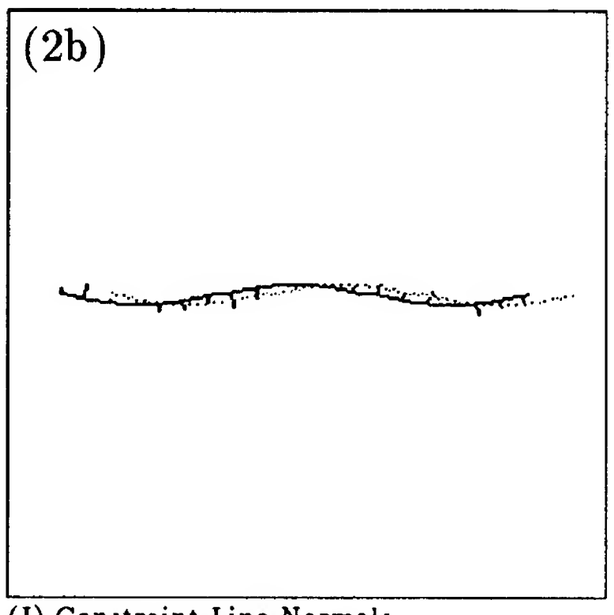
(I) Constraint Line Normals
(with 10% noise)



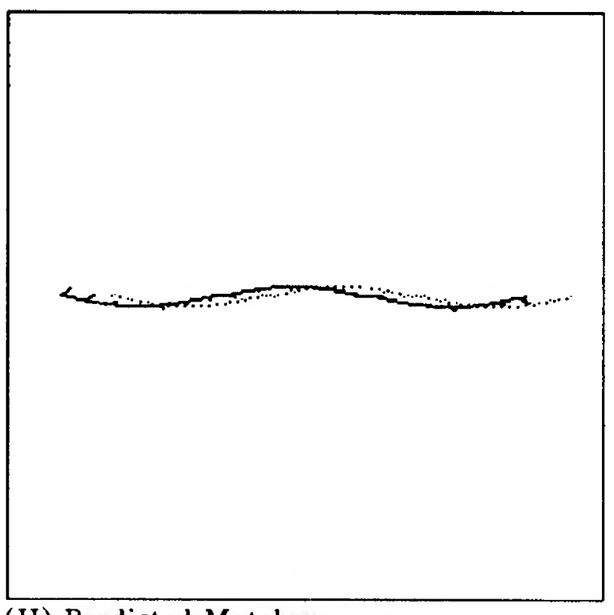
(II) Predicted Matches
(average percent error: 2.18%)



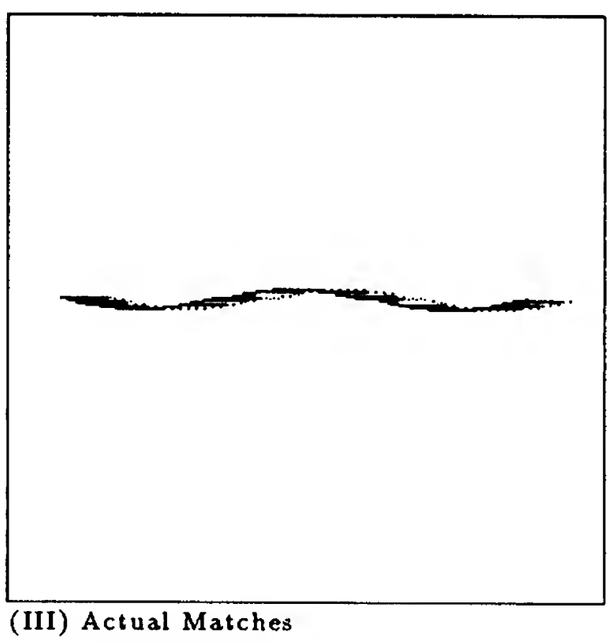
(III) Actual Matches



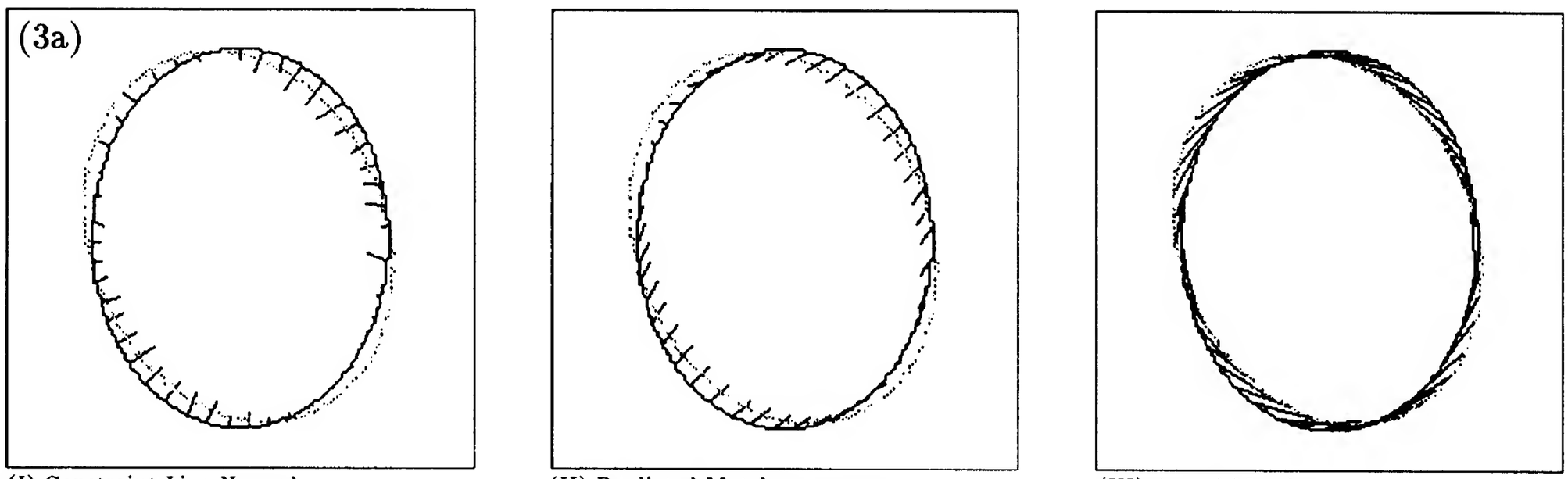
(I) Constraint Line Normals
(with 10% noise)



(II) Predicted Matches
(average percent error: 87.0%)



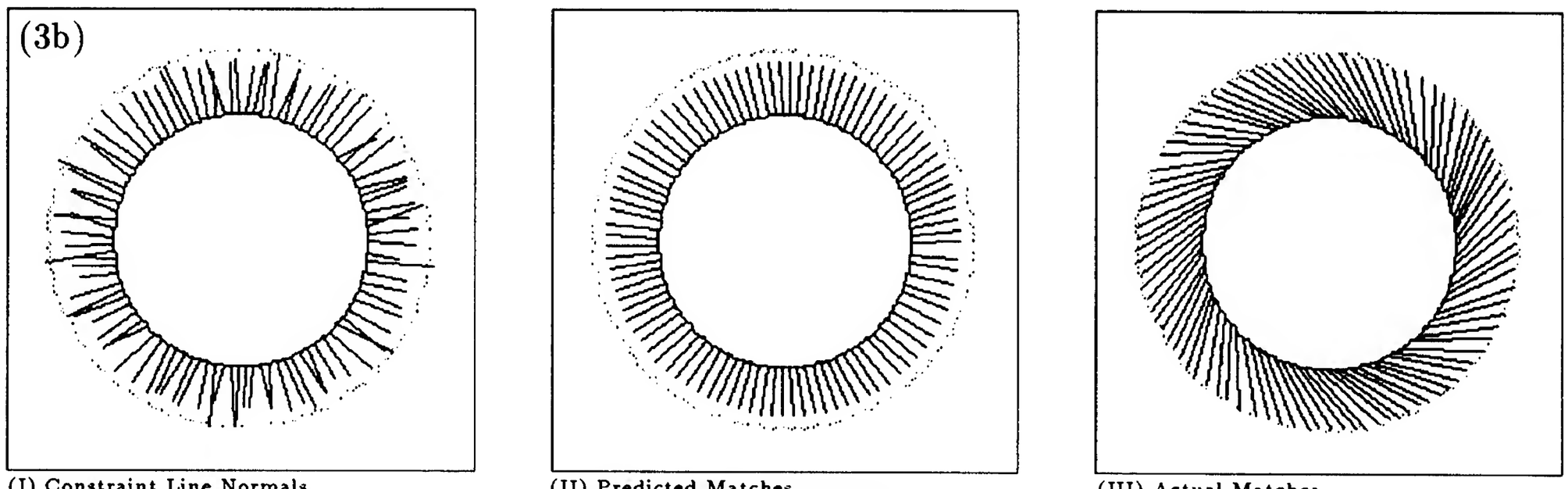
(III) Actual Matches



(I) Constraint Line Normals
(with 10% noise)

(II) Predicted Matches
(average percent error: 98.68%)

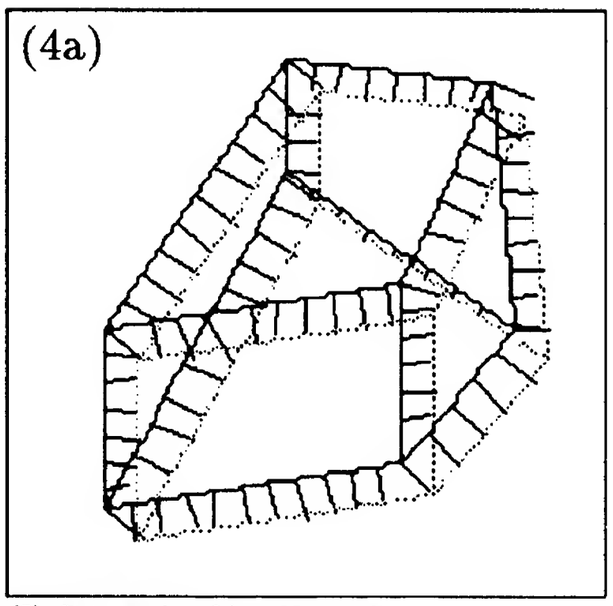
(III) Actual Matches



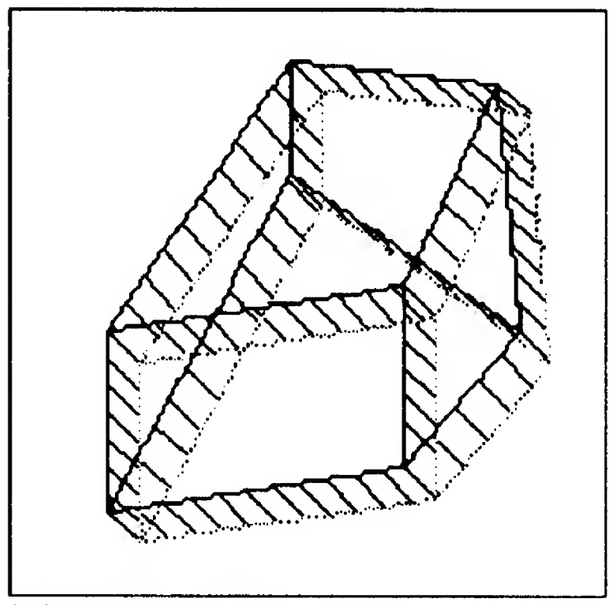
(I) Constraint Line Normals
(with 10% noise)

(II) Predicted Matches
(average percent error: 76.13%)

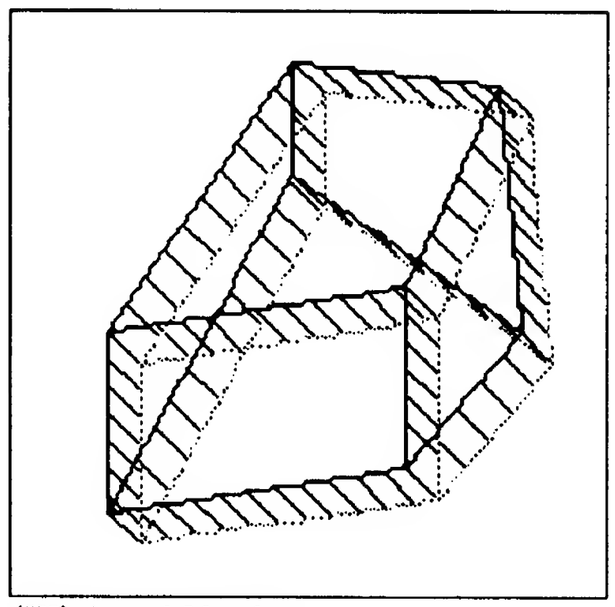
(III) Actual Matches



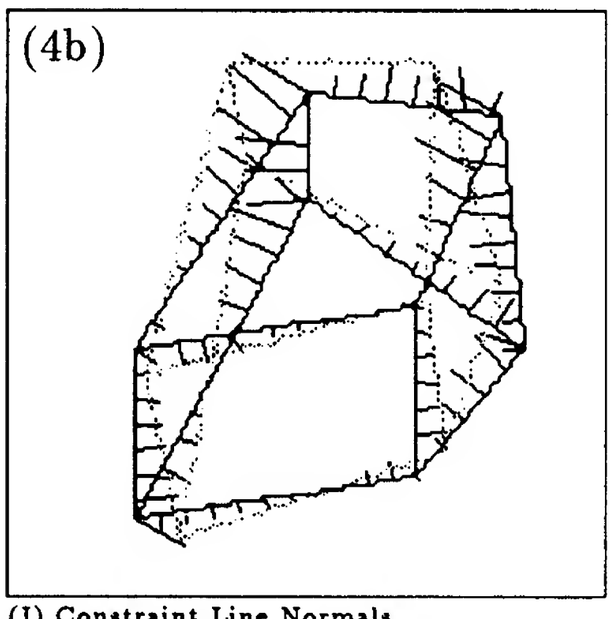
(I) Constraint Line Normals
(with 10% noise)



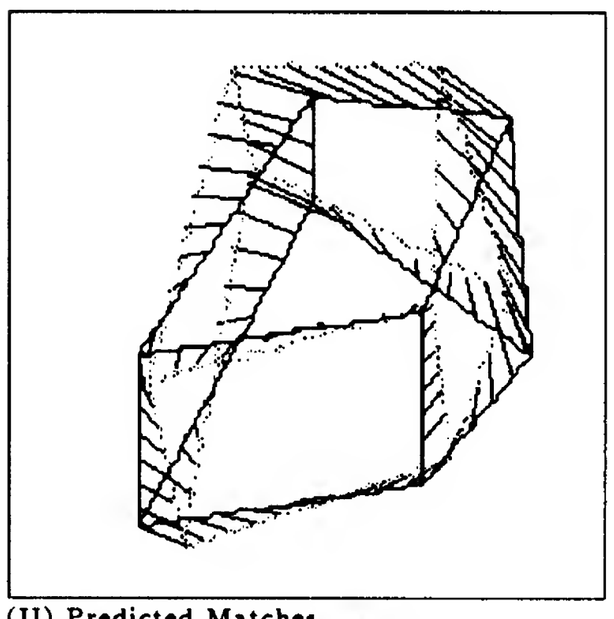
(II) Predicted Matches
(average percent error: 3.43%)



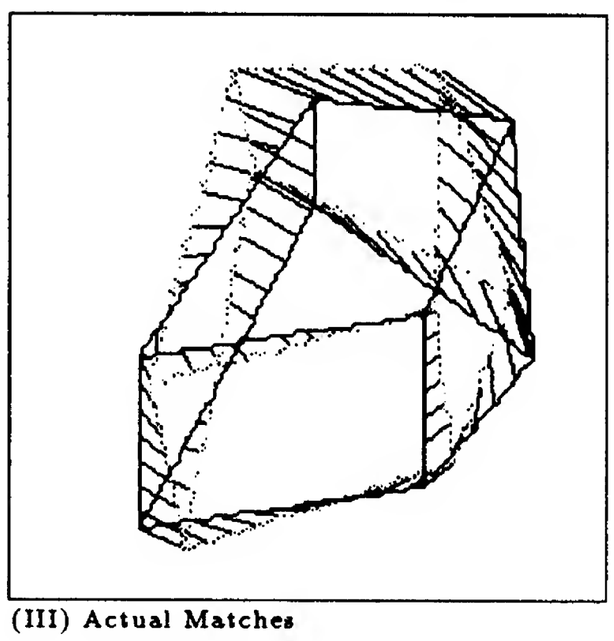
(III) Actual Matches



(I) Constraint Line Normals
(with 10% noise)



(II) Predicted Matches
(average percent error: 7.80%)

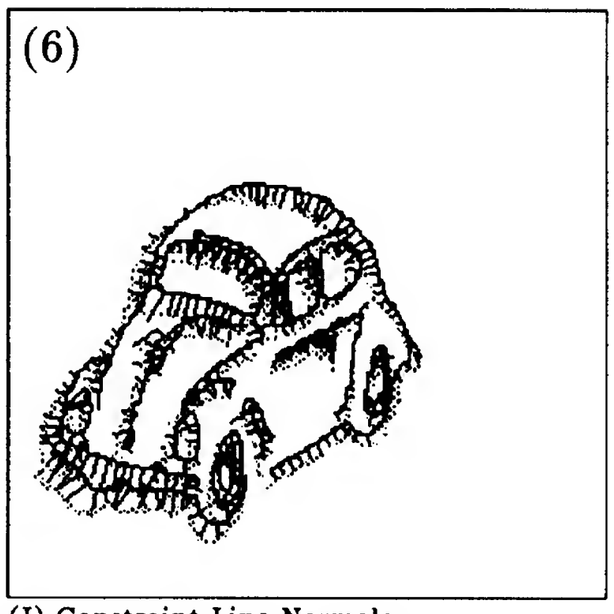


(III) Actual Matches

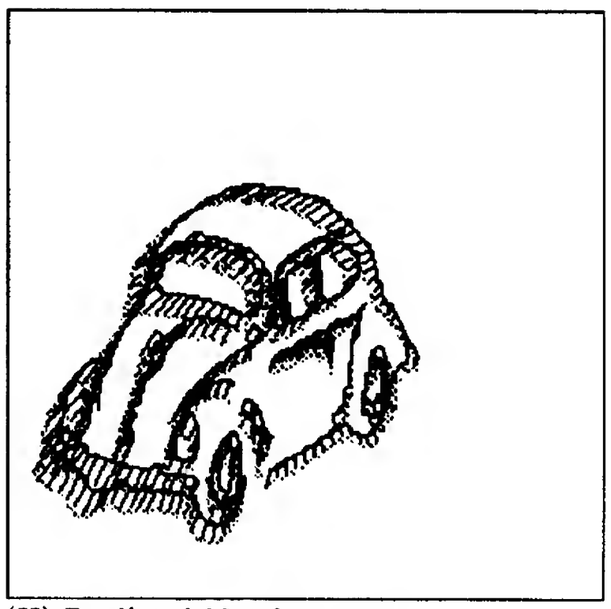
(I) Constraint Line Normals
(with 10% noise)

(II) Predicted Matches
(average percent error: 9.80%)

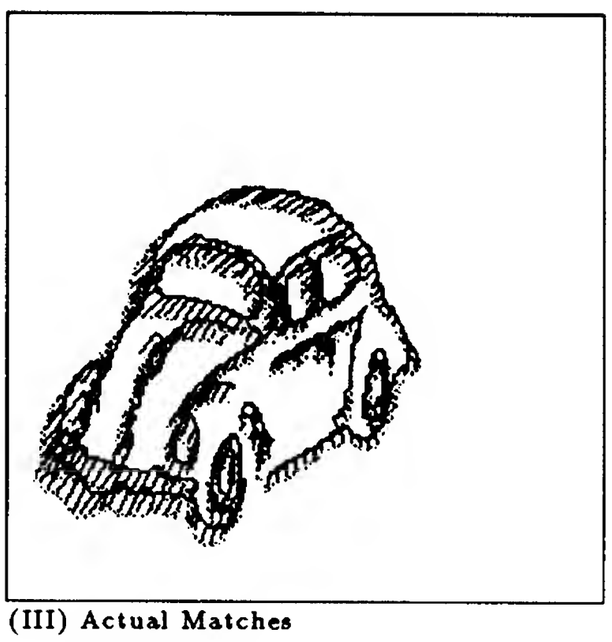
(III) Actual Matches



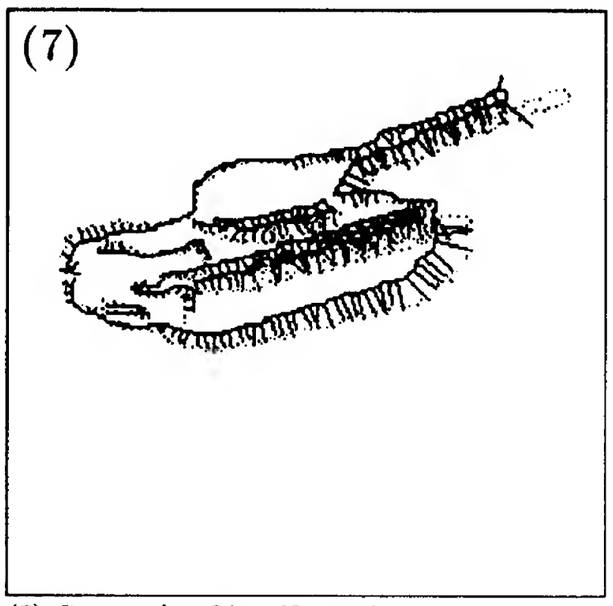
(I) Constraint Line Normals
(with 10% noise)



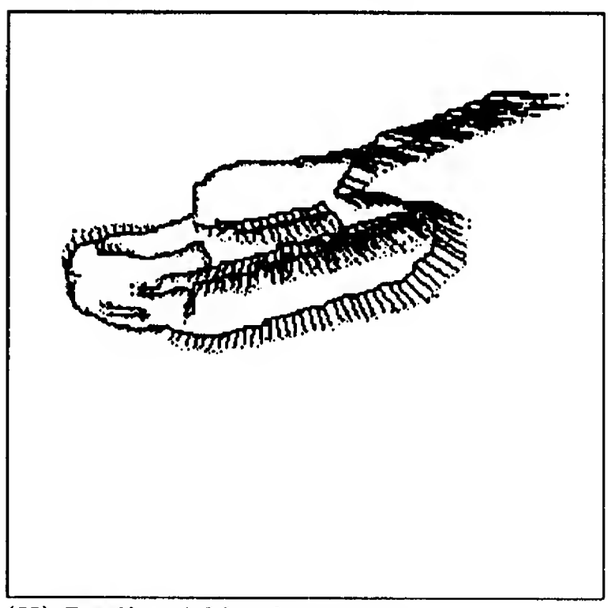
(II) Predicted Matches
(average percent error: 7.82%)



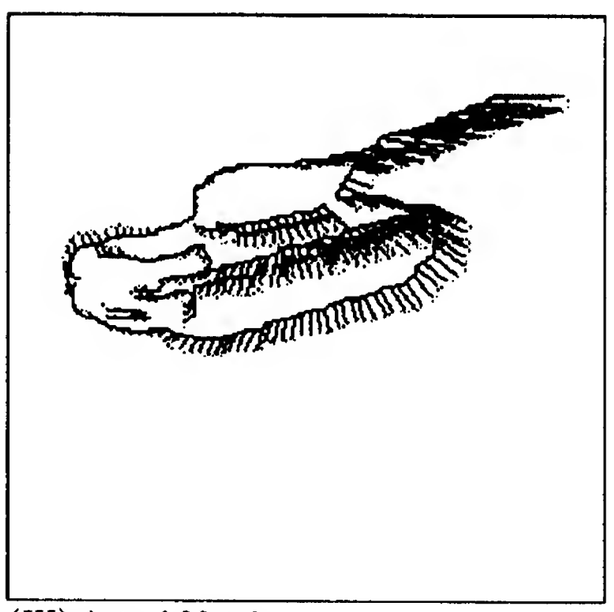
(III) Actual Matches



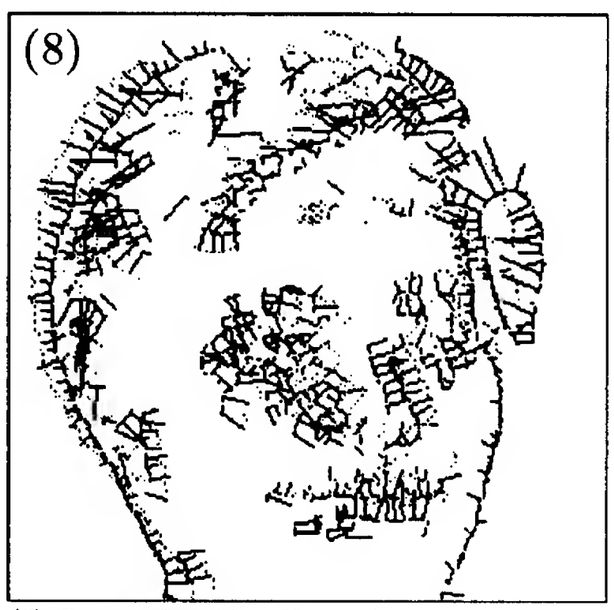
(I) Constraint Line Normals
(with 10% noise)



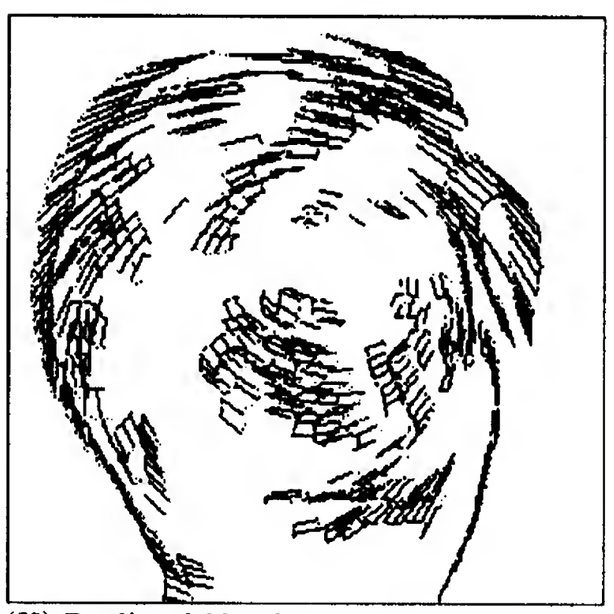
(II) Predicted Matches
(average percent error: 9.98%)



(III) Actual Matches

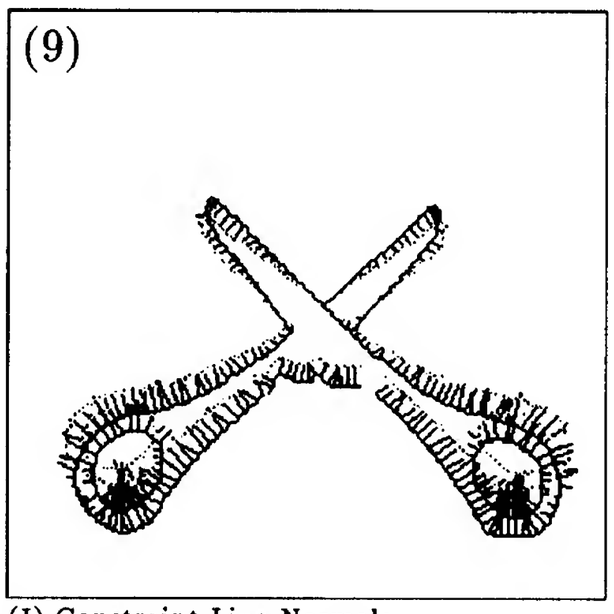


(I) Constraint Line Normals
(with 10% noise)

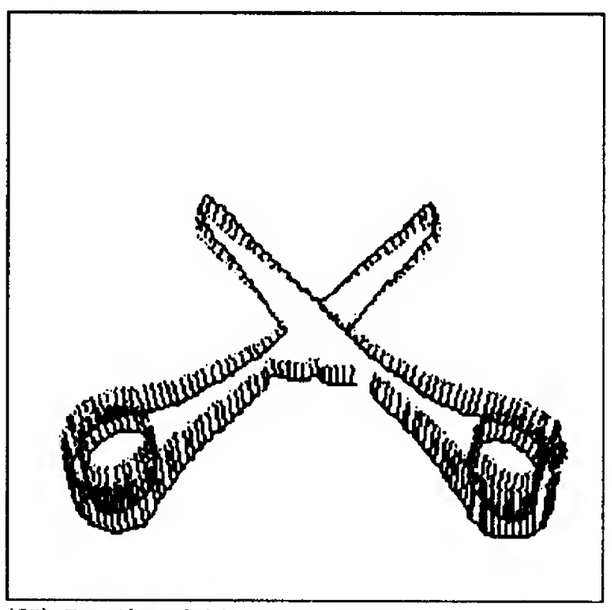


(II) Predicted Matches
(average percent error: 7.58%)

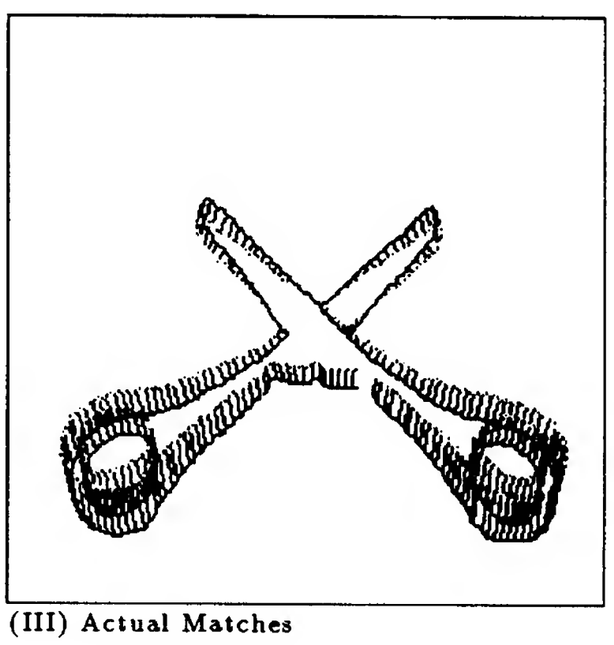
(III) Actual Matches



(I) Constraint Line Normals
(with 10% noise)



(II) Predicted Matches
(average percent error: 8.49%)



(III) Actual Matches

degrees about each axis, translated, scaled by a factor of 1.2, and stretched by a factor of 1.1 along each axis. Finally, the matching of the edges obtained from a pair of roughly aligned natural views of a Volkswagen (borrowed from [5]), a tank, a doll face (borrowed from [37]), and a pair of scissors (borrowed from [5]) are respectively presented in rows 6 through 9.

B Modified Pseudoinverse

This appendix describes two modifications to the general pseudoinverse. The general pseudoinverse is defined for any $n \times m$ matrix A. First, A is uniquely diagonalized as

$$A = Q_1 \Lambda Q_2^T \tag{57}$$

where Q_1 and Q_2 are $n \times m$ and $m \times n$ orthonormal matrices respectively, Λ is the $m \times n$ matrix of the singular values of A,

$$\Lambda = \begin{bmatrix} \sqrt{\lambda_1} & & 0 & 0 & \cdots & 0 \\ & \ddots & & \vdots & & \vdots \\ 0 & & \sqrt{\lambda_n} & 0 & \cdots & 0 \end{bmatrix} \tag{58}$$

(in this particular case for $n \leq m$), and $\lambda_1 \cdots \lambda_n$ are the eigenvalues of $A^T A$. This process is called *singular value decomposition* (SVD). Given this diagonalization, the general form of the pseudoinverse is

$$A^+ = Q_2 \Lambda^+ Q_1^T \tag{59}$$

where

$$\Lambda_{ij}^+ = \begin{cases} 1/\Lambda_{ij} \text{ if } \Lambda_{ij} \geq t \\ 0 \text{ otherwise} \end{cases} \tag{60}$$

and t is a threshold below which values are effectively singular. The higher the threshold, the more stable, but the less precise, the result. To regulate stability, we choose this threshold such that the condition number of $A^T A$ is less than some number, κ_{max}. Thus, our threshold is

$$t = \sqrt{\frac{\lambda_{max}}{\kappa_{max}}} \tag{61}$$

since the condition number is the ratio of the maximum to minimum eigenvalues. This is the first modification.

The problem with this formulation of the general pseudoinverse is that the solution suddenly becomes singular when A is not full rank. This discontinuity, due to a very small change in A, is quite sharp because the elements of Λ^+ do not continuously vary with the elements of Λ. Consider instead that the elements of Λ^+ gradually default to zero using:

$$\Lambda_{ij}^+ = \begin{cases} 1/\Lambda_{ij} \text{ if } \Lambda_{ij} \geq t \\ \Lambda_{ij}/t^2 \text{ otherwise} \end{cases} . \tag{62}$$

Now the elements of Λ^+ are continuous functions of the elements of Λ. This is the second modification. (See [38] for a more detailed presentation of SVD and the general pseudoinverse.)

References

[1] E. H. Adelson and J. A. Movshon. Phenomenal coherence of moving visual patterns. *Nature*, 300:523–525, 1982.

[2] P. Anandan and R. Weiss. Introducing a smoothness constraint in a matching approach for the computation of optical flow fields. In *IEEE Workshop on Computer Vision: Representation and Control*, pages 186–194, Bellaire, MI, October 1985.

[3] K. Arbter. Affine-invarient fourier descriptors. In *COST 13 Workshop*, pages 22–27, Bonas, France, August 1988.

[4] I. A. Bachelder. Contour matching using local affine transformations. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, June 1991.

[5] R. Basri. *Recognition of 3-D solid objects from 2-D images.* PhD thesis, Weizmann Institute, Rehovot, Isreal, October 1990.

[6] P. J. Burt, J. Bergen, R. Hingorani, S. Peleg, and P. Anandan. Dynamic analysis of image motion for vehicle guidance. In *IEEE International Workshop on Intelligent Motion Control*, pages 75–82, Bogazici University, Istanbul, August 20-22 1990.

[7] P. J. Burt, J. R. Bergen, R. Hingorani, R. Kolczynski, W. A. Lee, A. Leung, J. Lubin, and H. Shvaytser. Object tracking with a moving camera. In *Proceedings of the workshop on visual motion*, pages 2–12, Irvine, CA, March 20-22 1989.

[8] D. Cyganski and J. A. Orr. Applications of tensor theory to object recognition and orientation determination. *IEEE Trans Patt Anal Mach Intell*, PAMI-7(6):662–673, 1985.

[9] O. D. Faugeras and M. Hebert. The representation, recognition, and location of 3-d objects. *Int. J. of Robotics Res.*, 5:27–52, 1986.

[10] C. L. Fennema and W. B. Thompson. Velocity determination in scenes containing several moving objects. *Comput. Graph. Image. Proc.*, 9:301–315, 1979.

[11] J. J. Gibson and E. J. Gibson. Continuous perspective transformations and the perception of rigid motion. *Journal of Experimental Psychology*, 54:129–138, 1957.

[12] E. C. Hildreth. *The Measurement of Visual Motion.* MIT Press, Cambridge, 1984a.

[13] E. C. Hildreth. The computation of the velocity field. *Proc. R. Soc. London B*, 221:189–220, 1984b.

[14] E. C. Hildreth and S. Ullman. The computational study of vision. A.I. Memo 1038, The Artificial Intelligence Lab., M.I.T., April 1988.

[15] J. Hong and X. Tan. The similarity between shapes under affine trasformation. Robotics Res. Rep 133, New York University, December, 1987.

[16] B. K. P. Horn. *Robot Vision.* The MIT Press and McGraw-Hill, Cambridge and NY, 1986.

[17] B. K. P. Horn and B. G. Schunk. Determining optical flow. *Artif. Intell.*, 17:185–203, 1981.

[18] D. P. Huttenlocher and S. Ullman. Recognizing solid objects by alignment with an image. *International Journal of Computer Vision*, 5(2):195–212, 1990.

[19] J. J. Koenderink and A. J. Van Doorn. Local structure of movement parallax of the plane. *Journal of the Optical Society of America*, 66:717–723, 1976.

[20] Y. Lamdan, J. T. Schwartz, and H. J. Wolfson. Affine invariant model-based object recognition. *IEEE Transactions on Robotics and Automation*, 6(5):578–589, 1990.

[21] J. S. Lappin and H. H. Bell. The detection of coherence in moving random dot patterns. *Vision Research*, 16:161–168, 1976.

[22] D. G. Lowe. *Perceptual Organization and Visual Recognition.* Kluwer Academic, Boston, 1985.

[23] D. Marr and E. C. Hildreth. Theory of edge detection. *Proc R Soc. London B*, 207:187–217, 1980.

[24] D. Marr and T. Poggio. Cooperative computation of stereo disparity. *Science*, 194:283–287, 1976.

[25] D. Marr and S. Ullman. Directional selectivity and its use in early visual processing. *Proc R Soc. London B*, 211:151–180, 1981.

[26] J. E. W. Mayhew and J. P. Frisby. Psychophysical and computational studies towards a theory of human stereopsis. *Artificial Intelligence*, 17:349–385, 1981.

[27] H. H. Nagel. Recent advances in image sequence analysis. In *Proc. Premier Colloque Image - Traitement, Synthese, Technologie et Applications*, pages 545–558, Biarritz, France, May 1984.

[28] H. H. Nagel and W. Enkelmann. Towards the estimation of displacement vector fields by "oriented smoothness" constraints. In *7th Int. Conf. on Pattern Recognition*, pages 6–8, Montreal, Canada, July 1984.

[29] K. Nakayama. Biological motion processing: A review. *Vision Research*, 25:625–660, 1985.

[30] K. Nakayama and G. H. Silverman. The aperture problem-I. Perception of nonrigidity and motion direction in translating sinusoidal lines. *Vision Research*, 28:739–746, 1987a.

[31] K. Nakayama and G. H. Silverman. The aperture problem-II. Spacial integration of velocity information along contours. *Vision Research*, 28:747–753, 1987b.

[32] A. J. Pantle and L. Picciano. A multistable display: Evidence for two separate motion systems in human vision. *Science*, 193:500–502, 1976.

[33] G. F. Poggio and T. Poggio. The analysis of stereopsis. *Annual Reviews of Neuroscience*, 7:379–412, 1984.

[34] D. Regan and K. I. Beverley. Visual fields for frontal plane motion and changing size. *Vision Research*, 23:673–676, 1983.

[35] H. A. Saito, M. Yukie, K. Tanaka, K. Hikosaka, Y. Fukuda, and E. Iwai. Interaction of direction signals of image motion in the superior temporal sulcus of the macaque monkey. *J. Neurosci.*, 6:145–157, 1986.

[36] R. Sekuler, S. Anstis, O. J. Braddick, T. Brandt, J. A. Movshon, and G. Orban. The perception of motion. In Lothar Spillman and John S. Werner, editors, *Visual Perception: The Neurophysiological Foundations.* Academic Press, 1990.

[37] A. Shashua. Illumination and 3D object recognition. In John E. Moody, Steve Hanson, and Richard Lippmann, editors, *Advances in neural information processing systems 4*. Morgan Kaufmann, 1992 (in press). Proc. NIPS '91, Denver CO.

[38] G. Strang. *Introduction to Applied Mathematics.* Wellesley-Cambridge Press, Wellesly, MA, 1986.

[39] J. T. Todd and P. Bressan. The perception of 3-dimensional affine structure from minimal apparent motion sequences. *Perception and Psychophysics*, 48(5):419–430, 1990.

[40] S. Ullman. Aligning pictorial descriptions: An approach to object recognition. *Cognition*, 32(3):193–254, 1989.

[41] J. P. H. Van Santen and G. Sperling. A temporal covariance model of motion perception. *J. Opt. Soc. Am. A*, 1:451–473, 1984.

[42] T. Wakahara. On-line cursive script recognition using local affine transformation. *Syst. Comput. Jpn. (USA)*, 20(7):10–19, 1988.

[43] A. M. Waxman and S. Ullman. Surface structure and three-dimensional motion from image flow kinematics. *International Journal of Robotics Research*, 4:72–94, 1985.

[44] A. M. Waxman and K. Wohn. Contour evolution, neighborhood deformation, and global image flow: Planar surfaces in motion. *International Journal of Robotics Research*, 4:95–108, 1985.

[45] D. W. Williams and R. Sekuler. Coherent global motion percepts from stochastic local motions. *Vision Research*, 24:55–62, 1984.